# BEST AVAILABLE IMAGES

Defective images within this document are accurate representations of the original documents submitted by the applicant..

Defects in the images may include (but are not limited to):

- BLACK BORDERS
- TEXT CUT OFF AT TOP, BOTTOM OR SIDES
- FADED TEXT
- ILLEGIBLE TEXT
- SKEWED/SLANTED IMAGES
- COLORED PHOTOS
- BLACK OR VERY BLACK AND WHITE DARK PHOTOS
- GRAY SCALE DOCUMENTS

## IMAGES ARE BEST AVAILABLE COPY.

**As rescanning documents *will not* correct images, please do not report the images to the Image Problem Mailbox.**

PCT WORLD INTELLECTUAL PROPERTY ORGANIZATION
International Bureau

INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(51) International Patent Classification 5 :
A61M 15/00, 16/10, A62B 7/00
F16K 31/02

A1

(11) International Publication Number: WO 92/12750

(43) International Publication Date: 6 August 1992 (06.08.92)

(21) International Application Number: PCT/US92/00566

(22) International Filing Date: 22 January 1992 (22.01.92)

(30) Priority data:
645,579 24 January 1991 (24.01.91) US

(71) Applicant: VORTRAN MEDICAL TECHNOLOGY, INC. [US/US]; 3941 J Street, Suite 354, Sacramento, CA 95819-3633 (US).

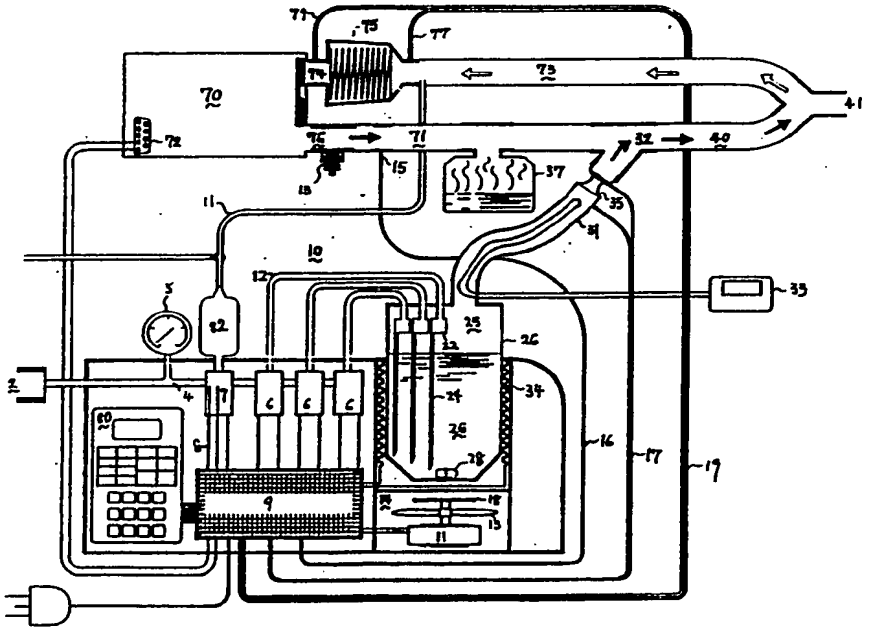
(72) Inventors: RAABE, Otto, G. ; 652 Buchanan Street, Davis, CA 95616 (US). LEE, James, I., C. ; 7861 Rush River Drive, Sacramento, CA 95831 (US). HATHAWAY, James, Calvin ; 441 Hartnell Place, Sacramento, CA 95825 (US).

(74) Agents: YIN, Ronald, L. et al.; Limbach & Limbach, 2001 Ferry Building, San Francisco, CA 94111 (US).

(81) Designated States: AT (European patent), AU, BE (European patent), CA, CH (European patent), DE (European patent), DK (European patent), ES (European patent), FR (European patent), GB (European patent), GR (European patent), IT (European patent), JP, LU (European patent), MC (European patent), NL (European patent), NO, SE (European patent).

**Published**
*With international search report.*
*Before the expiration of the time limit for amending the claims and to be republished in the event of the receipt of amendments.*

(54) Title: INTERMITTENT SIGNAL ACTUATED NEBULIZER SYNCHRONIZED WITH EXHALATION



(57) Abstract

A self-contained, high capacity nebulizer (10), having automatic mixing (28) and temperature control (34) features is provided. The nebulizer is designed for use in conjunction with mechanical respirators (70), ventilators, or breathing machines, and for this purpose will use electrical signals (8) generated by or received from the respirator (70) to automatically control and synchronize the nebulizing and mixing functions such that nebulization occurs only during the exhalation phase of the respiratory function to load the gas passageway of the respirator (70) to the patient with a standardized dose of medicinal aerosol. Upon commencement of the inhalation phase, the aerosol in the gas passageway is ventilated into the lungs of the patient to which it is attached.

*FOR THE PURPOSES OF INFORMATION ONLY*

Codes used to identify States party to the PCT on the front pages of pamphlets publishing international applications under the PCT.

| | | | | | |
|---|---|---|---|---|---|
| AT | Austria | FI | Finland | ML | Mali |
| AU | Australia | FR | France | MN | Mongolia |
| BB | Barbados | GA | Gabon | MR | Mauritania |
| BE | Belgium | GB | United Kingdom | MW | Malawi |
| BF | Burkina Faso | GN | Guinea | NL | Netherlands |
| BG | Bulgaria | GR | Greece | NO | Norway |
| BJ | Benin | HU | Hungary | PL | Poland |
| BR | Brazil | IE | Ireland | RO | Romania |
| CA | Canada | IT | Italy | RU | Russian Federation |
| CF | Central African Republic | JP | Japan | SD | Sudan |
| CG | Congo | KP | Democratic People's Republic of Korea | SE | Sweden |
| CH | Switzerland | | | SN | Senegal |
| CI | Côte d'Ivoire | KR | Republic of Korea | SU | Soviet Union |
| CM | Cameroon | LI | Liechtenstein | TD | Chad |
| CS | Czechoslovakia | LK | Sri Lanka | TG | Togo |
| DE | Germany | LU | Luxembourg | US | United States of America |
| DK | Denmark | MC | Monaco | | |
| ES | Spain | MG | Madagascar | | |

# INTERMITTENT SIGNAL ACTUATED NEBULIZER SYNCHRONIZED WITH EXHALATION

This application is a continuation-in-part of copending U.S. Patent Application Serial No. 07/585,616, filed on September 20, 1990, which is a continuation of U.S. Patent Application Serial No. 270,520, filed on November 14, 1988, now abandoned, which is a continuation of U.S. Patent Application Serial No. 07/071,202, filed on July 8, 1987, now U.S. Patent 4,832,012.

## Technical Field

The present invention relates to nebulizers for creating medicinal aerosols for inhalation therapy. In particular, the present invention relates to nebulizers used during the exhalation phase of the breathing cycle in conjunction with and without interfering with mechanical breathing machines which are used to ventilate the lungs of patients who cannot breathe unaided.

## Background Art

The thin membrane of the lungs provides an easily penetrated, convenient and generally safe means for obtaining rapid absorption of medication by the body. This is especially desirable where the lungs themselves are diseased or injured. Such medication or drugs are generally delivered to the lung membrane in the form of a fine mist or aerosol which is breathed into the lungs through the nose or mouth of the patient. A variety of devices, called nebulizers by those skilled in the art, have been developed for converting liquids into fine aerosols

for this purpose. The simplest of these devices is the hand-held atomizer which converts a liquid to an aerosol when a bulb is compressed to produce a jet of air which atomizes the medication and propels it out of the atomizer. To be effective, the aerosols need to be provided at high concentrations and with droplet size in the respirable range (mass median aerodynamic diameter less than 3 micrometers).

Nebulizers are particularly useful for initiating and continuing respiratory therapy in conjunction with respirators, mechanical ventilators or breathing machines (hereinafter referred to generically as respirators) used to ventilate the lungs of patients having serious respiratory impairment. While some respirators incorporate nebulizers in their design, many do not. Nebulizers incorporated into the structure of such respirators often suffer from many disadvantages. One such disadvantage is severely limited capacity for medication to be nebulized, requiring frequent interruptions in the therapy as new medication is added to the nebulizer reservoir.

Most, if not all, such nebulizers are incorporated in respirators in which the inhalation and exhalation phases of the breathing cycle are triggered by changes in air pressure caused by the patient himself. Such "demand" respirators are not useful for patients whose respiratory systems are paralyzed and incapable of causing even slight changes in air pressure. These patients are aided by mechanical respirators in which the phases of the breathing cycle are triggered by electrical signals. There is now no effective means for patients on such respirators to receive aerosol treatment.

Thus, the need exists for a nebulizer which can be attached to a mechanical respirator, especially those in which the breathing cycle is controlled by an electrical signal, which has a reservoir capacity sufficient to enable several hours of continuous treatment, which can prevent the settling of suspensions or mixtures without creating nebulization-destroying turbulence.

U.S. Patent 4,832,012 discloses the principal of signal actuated synchronization of nebulization for delivery of aerosolized medicine to patients whose breathing is supported or augmented by a mechanical respiratory. In that reference, nebulization could be effected during inhalation or exhalation, but the primary trust of that reference was to provide aerosols during the inhalation phase of the breathing cycle to mix with the inhalation tidal volume provided by the respirator, and in synchrony with the normal operation of the respiratory. However, it has been found that the addition of volume of gas to mix with the inhalation tidal volume provided by the respirator, may interfere with the normal operation of the respirator in certain operating modes, and the medicinal aerosol is diluted by the portion of gas delivered by the respirator.

Summary of the Invention

The present invention is based upon the nebulization of medicine during and synchronized with the exhalation portion of each breath of the breathing cycle to fill the airline leading from the nebulizer to the patient with a standardized dose of medicinal aerosols that are delivered to the lung by the force of the flow of breathing gas (oxygen-enriched air) delivered by the respirator during the

inhalation portion of the breathing cycle. One advantage of this invention is that more concentrated standardized dose of aerosol is delivered to the patient with the first parcel of gas that enters the lungs for each breath during the inhalation process. In addition, the signal used to actuate the nebulizer may be obtained from the ventilator or from an independently generated signal established by the nebulization system utilizing the readily detected respiratory air line pressure or pressure drop across filter from exhaled gas flow. Also, certain safety monitoring features are incorporated into such a system to detect aerosol clogging of respiratory filters and prevent interference with the normal operation of the respirator.

The nebulization system of the present invention can be attached to or operated with a mechanical respirator utilizing either a breathing cycle electrical signal obtained from the respiratory or an independent electrical signal generated by the nebulizer system which detects and responds to the exhalation initiation of the respirator. Such a synchronized signal actuated nebulizer system is designed to operate during the exhalation phase of the breathing cycle while treating a sick patient and efficiently providing, in the short time available, a medicinal aerosol in the appropriate and desired volume, concentration, and particle size distribution for deposition in the respiratory airways of the lungs. An important feature of such a system is that all of the aerosol is generated quickly (in about 1 second or less) and in a way that does not interfere with the control system of the respirator. The nebulizer system has a reservoir of capacity sufficient to enable several hours of continuous

treatment and with provision to prevent the settling of suspensions or mixtures without creating nebulization-destroying turbulence, and provides a precisely measured volume of medicinal aerosol generated during patient exhalation in a manner to reach the patient at the precise moment when inhalation begins.

In one embodiment, the present invention provides a nebulizer for use with mechanical respirators which use electrical signals to control the breathing cycle. The nebulizer of this embodiment uses the existing electrical signals from the mechanical respirator to synchronize aerosol generation to fill the gas passageway from the respirator to the patient during the exhalation cycle. Upon the initiation of the inhalation cycle, the aerosol is delivered from the gas passageway to the patient. Nebulization is obtained in this embodiment using the premixed oxygen-enriched air provided at high pressure to the respirator. Automatic temperature regulation and stirring of the liquid medication is optionally provided to preclude concentration change, separation or settling of the medication. Finally, a large volume reservoir is provided to eliminate the need for refilling during lengthy treatment protocols.

Brief Description of the Drawings

Figure 1 is a schematic side view of a nebulizer of the present invention operationally attached to a mechanical respirator;

Figure 2 is a perspective view of the intermittent signal actuated system of the present invention.

Detailed Description of the Invention

Figure 1 shows a nebulizer apparatus 10 of the present invention operably connected to a mechanical respirator 70. The nebulizer apparatus 10 comprises, in a housing, compressed gas inlet 2, at one end of a compressed gas conduit 4, adapted to be connected to a compressed gas source at pressure indicated by gauge 5. Preferably this compressed gas source is the same source which is furnishing oxygen-enriched air to the respirator 70, and provides compressed air or oxygen mixture to the nebulizer ranging up to about 50 psig.

Compressed gas conduit 4 is connected at the other end to a first electrically operated nebulizer valve 7, and a plurality of second electrically operated nebulizer valves 6, all of which are substantially similar. Examples of such valves which have been found useful include the Honeywell Skinner K4M ultraminiature 4-way solenoid operated pneumatic valve and Numatics HS series 2-way solenoid operated valves. Three valves 6 are shown in Figure 1.

Nebulizer valves 6 and 7 are connected by a plurality of electrical lead wires 8 to a microprocessor 9 and are controlled by the microprocessor 9. The microprocessor 9 receives the signals from a signal source 72 on the respirator 70 which controls the inhalation/exhalation phase of the breathing cycle. The microprocessor 9 controls the valves 6 and 7 to provide for a safe and effective operation. Examples of signal source 72 include a respirator solenoid, such as a solenoid actuated inhalation valve, an external electronic monitoring system, or an electronic interface attached to a signal generator on respirator 70, such as an

interface connected to a logic circuit in the respirator.

A control unit 80, whose control panel is shown in Figure 2, is connected to the microprocessor 9. The control unit 80 controls the functions of the nebulizing apparatus 10 of the present invention.

Each of the nebulizer valves 6 connects the compressed gas source 4 to nebulizer conduits 12 leading to aerosol nozzles 22. Each nebulizer valve 6 switches between two positions as electrical on/off signals are received. In the first position, during the exhalation phase of the respirator 70 when the electric signal is "on", a passageway is opened between compressed gas conduit 4 and nebulizer conduits 12 and remain open until the desired aerosol volume has generated or until the inhalation phase is initiated by the respiratory 70 as controlled by microprocessor 9. In the second position, when the electric signal is "off", the nebulizer conduits 12 are sealed off.

Nebulizer conduits 12 are attached at their other ends to aerosol nozzles 22, which include liquid feed tubes 24 extending into reservoir 26. Reservoir 26 includes magnetic stirring bar 28 which is located in the bottom of the reservoir. The liquid medicine contained in reservoir 26 is preferably kept at constant temperature by a reservoir heater or cooler 34.

A chamber 14 houses an AC motor 11 which rotates a cooling fan 13 and a magnet 18. The rotation of the magnet 18 causes the stir bar 28 to rotate to prevent sedimentation or separation of medicinal constituents.

The liquid medicine in the reservoir 26 is drawn via the liquid feed tubes 24 and is converted by the

aerosol nozzles 22 into an aerosol having droplets with a mass median aerodynamic diameter less than about 3 micron. The aerosol is generated into the air space 25 above the reservoir 26. The aerosol generated in the air space 25 enters into an aerosol tube 31.

The temperature of the aerosol in the aerosol tube 31 is controlled by a temperature controller 33. In one embodiment, the temperature controller is simply an electric heater having a control unit. Within the aerosol tube 31 is also a neb-flow sensor 35. The neb-flow sensor 35 detects the amount of aerosol being delivered through the aerosol tube 31. The output of the neb-flow sensor 35 is supplied as a signal to the microprocessor 9 via neb-flow sensor pressure/vacuum lines 17.

The respirator 70 has an inhalation tube 71 and an exhalation tube 73. The inhalation tube 71 fluidically connects the respirator 70 to a patient and during the inhalation phase, breathing gas is supplied from the respirator 70 along the inhalation tube 71 into the respiratory tract of the patient. The aerosol tube 31 connects the air space 25 above the liquid 26 to the inhalation tube 71 at a nebulizer input 30. In addition, a pop-off valve 13 is also located in the inhalation tube 71. The function of the pop-off valve 13 is to relieve any pressure which is generated to dangerous levels within the inhalation tube 71. It functions purely as an emergency safety valve. Finally, an airway pressure sensor 15 is also positioned in the inhalation tube 71. The airway pressure sensor 15 generates a signal which is also supplied to the microprocessor 9 via airway pressure monitoring line 16. A humidifier 37 whose output is water vapor

mixed with the breathing gas is also connected to the inhalation tube 71.

The exhalation tube 73 fluidically connects the patient to the respirator 70. Located within the exhalation tube 73 is an exhalation filter 75. Upstream from the exhalation filter 75, i.e., between the exhalation filter 75 and the patient is an upstream filter pressure sensor 77. Downstream from the exhalation filter 75, i.e., between the exhalation filter 75 and the ventilator 70 is a downstream filter pressure sensor 79. The upstream filter pressure sensor 77 and the downstream filter pressure sensor 79 each provide a signal which is supplied to the microprocessor 9.

The solenoid 7 is also connected to receive gas from the gas conduit 4 and is adapted to supply gas to a decay flow line 11 to the exhalation tube 73, upstream from the upstream filter pressure sensor 77. Thus, the solenoid 7, when activated, provides a stream of compressed gas which is supplied into the exhalation tube 73, between the patient and the upstream filter pressure sensor 77. The function of the decay solenoid 7 is also controlled by the microprocessor 9.

The operation of the nebulizer apparatus 10 of the present invention will be understood as follows. The practitioner first determines the amount of volume per breath of the standardized dose of aerosol which is to be generated by the apparatus 10 of the present invention which is to be supplied to the inhalation tube 71. The amount is entered on the control unit 80. The microprocessor 9 receives the signal and based upon its knowledge of the gas pressure from the compressed gas conduit 4, and the cross-sectional area of each of nebulizing nozzles

22, the microprocessor 9 calculates the amount of time which the solenoids 6 would have to be activated in order to introduce the desired amount of aerosol into the inhalation tube 71. Alternatively, the signal from the neb-flow sensor 35 is used by the microprocessor 9 to turn off the nebulizer solenoids 6 when the desired charging volume has been generated.

When the mechanical respirator 70 begins the exhalation phase of the respiratory cycle, electrical signal 72 supplies the signal to the microprocessor 9. (As will be discussed hereinafter, a number of other signals are supplied to the microprocessor 9 to indicate the beginning of the exhalation cycle. These additional signals are used in the event the ventilator 70 cannot provide the electrical signal source 72 or is used as a safety backup to the electrical signal source 72.) When the mechanical respirator 70 begins the exhalation phase, the inhalation port 76 is closed. The exhalation port 74 is opened, opening the exhalation tube 73.

After the electrical signal source 72 generates the signal indicating the beginning of the exhalation phase, the microprocessor 9 activates the solenoids 6 to the three nebulizing nozzles 24. Thus, after the commencement of the exhalation phase, and after the detection of the electrical signal, maximum generation of the aerosol from the apparatus 10 commences and continues until the standardized volume or dose of aerosol has been generated. Compressed gas flows through the compressed gas conduit 4 into the three nebulizer conduits 12 and into the nozzles 22, which draw liquid via liquid feed tube 24 from the liquid reservoir 26. The aerosol is then generated and is supplied into the air space 25 above

the reservoir 26. The aerosol generated in the air space 25 then enters into the aerosol tube 31 where the temperature thereof is controlled by the temperature controller 33. The aerosol then leaves the aerosol tube 31 and enters into the inhalation tube 71 through port 30. Generation of the standardized dose of aerosol fills the charging volume space 40 between the nebulizer input port 32 and the patient 41 in the inhalation tube 71. Any excessive aerosol will enter the exhalation tube 73 and return to the respirator 70.

During the exhalation phase, the pressure in the inhalation tube 71 is monitored by the airway pressure sensor 15 and is supplied to the microprocessor 9. This provides a safety signal to the microprocessor 9 to shut off the function of the aerosolization in the event pressure within the inhalation tube 71 builds to an excessive level or if inhalation begins. In addition, a mechanical safety pop-off valve 13 is provided wherein in the event the pressure in the inhalation tube 71 exceeds the pressure regulation of the pop-off valve 13, the valve 13 would automatically open relieving the pressure in the inhalation tube 71.

During the exhalation cycle, the respirator 70 continuously monitors the pressure on the exhalation tube 73. In order to provide for a smooth decay flow of gas entering into the exhalation tube 73 from the patient, and thereby simulating smooth exhalation reduction from the patient, the solenoid 7 is activated during the exhalation cycle. When the solenoid 7 is activated, the gas from the compressed gas conduit 4 fills a fixed volume chamber 82. The fixed volume chamber 82 has a calibrated orifice which is connected to the decay flow line 11 and is

supplied to the exhalation tube 73. During the time period in which the aerosol is being generated, the fixed volume chamber 82 is filled with breathing gas to a predetermined pressure. At the end of the charging period, the compressed gas from the gas conduit 4 is turned off. The gas from the fixed volume chamber 82 is then allowed to flow in a decay manner into the exhalation tube through the orifice connecting the chamber 82 to the decay flow line 11. When the pressure in the fixed chamber 82 gradually reduces, the flow entering the decay flow line 11 simulates a natural first order decay.

Synchronous with the beginning of the exhalation cycle, the three nebulizing nozzles 22 are turned on simultaneously or one at a time to produce the desired charging volume during a portion of the exhalation period to allow the respirator 70 to maintain and/or support the patient's spontaneous breathing effort without interference from the charging flow.

When the respirator 70 begins the inhalation phase of the respiratory cycle, the electrical signal source 72 switches to an "off" position. In the "off" position, the respirator inhalation port 76 opens; the respirator exhalation port 74 is closed.

The solenoid valves 6 are controlled by microprocessor 9 when first, the desired standardized dose is reached (usually only takes a portion of the exhalation phase), or secondly when microprocessor 9 detects the electrical signal source 72 turn to an "off" position. In the first priority, the solenoids 6 can be turned off one at a time. In the second case, the solenoids 6 are turned off immediately to allow respirator 70 to begin the inhalation phase.

The gradual turning off of the plurality of solenoids 6 generates a gradual pressure reduction and flow shaping that prevents spurious triggering of the respiratory ventilator 70 caused by rapid flow changes. Because the aerosol generated by the apparatus 10 of the present invention fills the inhalation tube 71 between the nebulizer input 30 and the patient with the desired standardized volume or aerosol dose, when the ventilator 70 begins the inhalation phase and pushes the gas in the inhalation tube 71 into the respiratory tract of the patient, the aerosol in the charging volume space 40 would be the first gas pushed into the lungs of the patient. Thus, the medicine produced by the aerosol would be first delivered to the patient during the inhalation cycle.

The advantage of the apparatus 10 and method of the present invention is that generating the aerosol and introducing it into the charging volume space 40 during the exhalation phase means the aerosol is pre-charged in the inhalation tube. Further, the amount of aerosol in the charging volume space 40 can be metered or controlled by the microprocessor 9. In addition, the introduction of aerosol during the exhalation phase does not perturb the pressure of the gas from the respirator 70 delivered during the inhalation phase.

As previously discussed, the source of electrical signal 72 may not be provided by all ventilators 70. The upstream filter sensor 77 and the downstream filter sensor 79 each provides a signal via the exhalation filter sensor pressure/vacuum lines 19, the difference of which indicates the commencement of the exhalation phase. Thus, upon the immediate commencement of the

exhalation phase, a pressure differential would be detected between the upstream filter sensor 77 and the downstream filter sensor 79, respectively. This pressure differential, supplied as a signal to the microprocessor 9, would indicate to the microprocessor 9 that the exhalation cycle has commenced. This signal can be used by microprocessor 9 to begin nebulization when no respirator electrical signal is available. Alternatively, the airway pressure sensor 15 supplies a signal to the microprocessor 9 indicating the beginning of the exhalation and also the beginning of the inhalation for control of the nebulization by microprocessor 9 when no respirator electrical signal is available.

In addition, there are many safety considerations with the apparatus 10 of the present invention. With the upstream and downstream filter sensor 77 and 79 respectively having an exhalation filter 75 therebetween, the condition of the exhalation filter 75 can be continuously checked. As the apparatus 10 of the present invention is continuously used, and as the filter 75 becomes increasingly clogged, the pressure differential between the upstream filter sensor 77 and the downstream filter sensor 79 would increase. Alternatively, the loading/clogging of the exhalation filter can be detected using the airway pressure sensor 15 which supplies a signal to microprocessor 9 via line 16. This is because airway pressure during nebulization is a function of the resistance of the exhalation filter. The filter loading/clogging can be detected by the microprocessor 9 and can be signaled on the control unit 80 as an alarm that the exhalation filter 75 needs to be examined and/or changed.

As previously discussed, the airway pressure sensor 15 provides an independent airway pressure measurement upstream to exhalation filter to monitor the patients safety. Finally, the control unit 80 can control the apparatus 10 to cause it to pause its operation. This provides an independent check on the respirator system 70. The control unit shown in Figure 2 provides for setting of charging volume, respirator selection (for different commercial respirators), heater temperature, nebulizer hold option, alarm test option, alarm reset, and alarm silence. Further, the control unit displays respirator selection, charging volume, alarm, warning, and caution, indication of exhalation filter loading, patient peak inspiratory pressure, heater temperature and nozzle gas pressure. Signals from the neb-flow sensor 35 are used to alarm if either inadequate charging volume is generated or if the nebulizer nozzle 24 malfunction in the "on" position. The microprocessor 9 provides yet additional safe and effective operation for the apparatus 10 of the present invention. In the preferred embodiment, the microprocessor 9 is an Intel 8751 available from Intel Corporation. A copy of the program, written in the assembly language, for execution by the microprocessor 9 is attached as Exhibit A.

```
                    :VISAN9A
                $TITLE SAMPLE SIGNALS AND CONTROL VISAN 9
                :SAMPLE VENTILATOR ANALOG SIGNAL AND
                ;PRESSURE AND FLOW SIGNALS FROM NEBULIZER
                :AND CONTROL 3 NEBULIZER VALVES.
                :CONTROL SERIAL INTERFACE WITH OPERATOR
                ;SWITCHES AND DISPLAYS.
                :
000B =          FLOTIM EQU 11   :TIME=2.2S
0032 =          NOFLOTIM EQU 50 :TIME=10S
002D =          FLO_TH EQU 45   :FLO 18LPM.0.14CMWC,0.17V,2DH
008C =          NOFLO_TH EQU 140 :FLO 35LPM,1.12CMWC.0.5V,8CH
00E0 =          PIP_THRESH SET 120*8/5+32 ;THR=4.4V,E0H,120CM
0037 =          FILTAWP_THRESH EQU 55 :PRES=34CM,1.07V,37H
008D =          FILTDP_THRESH EQU 141 :PRES=5.5CM,2.75V,8DH
0008 =          PATINSP_THRESH SET 5*8/5 ;PEEP-AWP= 5 CM WC
00A0 =          TEMP_HI SET 80*2 ;UPPER LIM 80C,A0H
                :
0000            FSEG
                                ;BANK0
0001 =          ALTNAME R1.RVENT_SIG ;VENTILATOR SIGNAL
0002 =          ALTNAME R2.RFLT_FLO ;EXH FILT DP SIGNAL
0003 =          ALTNAME R3,RAW_PRESS ;AWP TAP AT VENT
0004 =          ALTNAME R4,RNEB_FLO ;NEB OUTPUT DP
0005 =          ALTNAME R5.RTEMP ;TEMP DEG C * 2
0006 =          ALTNAME R6,RVENT  :VENTILATOR # SELECTED
                                ;BANK1
0001 =          ALTNAME R1,RCHG_TIM ;NEB CHARGE TIME
0002 =          ALTNAME R2,RDIV10 ;TIMER DIV BY 10
0003 =          ALTNAME R3,RDIV5  ;TIMER DIV BY 5
0004 =          ALTNAME R4,RON_TIM ;NEB FLOW ON TIME
0005 =          ALTNAME R5,ROFF_TIM ;NEB FLOW OFF TIME
0006 =          ALTNAME R6.RSIL_TIM ;AUDIO OFF TIME
0007 =          ALTNAME R7.RHOLD_TIM ;NEB OFF TIME
0000            ENDS
                ;
0000            DSEG
0023 =          LED1 DATA 23H :LED BANKS
0026 =          LED2 DATA 26H
0025 =          LED3 DATA 25H
0028 =          CHG_VOL DATA 28H :HUNS DEC DISPLAY
0029 =          DEC_HUN DATA 29H ':NUMBER FOR DISPLAY
002A =          DEC_TEN DATA 2AH
002B =          DEC_ONE DATA 2BH
002C =          FLTLD_HUN DATA 2CH ;FILTER LOAD SETTING
002D =          FLTLD_TEN DATA 2DH : 25%, 50% OR 75%
002E =          FLTLD_ONE DATA 2EH
002F =          THREE_CYCLE DATA 2FH :THREE BREATH COUNTS
0040 =          FLTFLO_LO DATA 40H :RUNNING AVG CALC
0044 =          FLTFLO_AVG DATA 44H
0045 =          CLOG_LO DATA 45H
0046 =          CLOG_HI DATA 46H
0048 =          AWP_LO DATA 48H
004C =          AWP_AVG DATA 4CH
004D =          AWP_MAX DATA 4DH
```

EXHIBIT A

```
004E =          PIP_STORE DATA 4EH
0050 =          POSSUM DATA 50H ;NEB POS SUM
0051 =          NEGSUM DATA 51H ;NEB NEG SUM
0055 =          FLTLD25 DATA 55H ;PERCENT FILTER LOAD
0056 =          FLTLD50 DATA 56H
0057 =          FLTLD75 DATA 57H
0058 =          PIP_LO DATA 58H
005B =          PIP_AVG DATA 5BH
0060 =          PEEP_LO DATA 60H
0063 =          PEEP_AVG DATA 63H
0011 =          TEMP_SET DATA 11H
0012 =          TEMP_DEC DATA 12H
0014 =          ONTIMER DATA 14H
0015 =          OFFTIMER DATA 15H
0019 =          SET_CHGTIM DATA 19H ;CONTROLS CHARGE VOL
001A =          VENT_LO DATA 1AH    ;LOWER THRESH
001B =          VENT_HI DATA 1BH    ;UPPER THRESH
001C =          TEMP_STORE DATA 1CH ;TEMPORARY STORE
001D =          DIVIDE1 DATA 1DH  ;TRANS_DEL
001E =          DIVIDE2 DATA 1EH
0068 =          VENT_LOW DATA 68H
006C =          VENT_AVG DATA 6CH
0028            ENDS
                ;
0000            BSEG
0000 =          WAIT BIT 0H ;FIVE BREATH WAIT
0001 =          EXH BIT 1H ;EXHALATION PERIOD
0002 =          DIV21 BIT 2H ;TIMER
0003 =          VOL_CHG BIT 3H ;OP CHANGING VOL SET
0004 =          VEN_SEL BIT 4H ;OP SELECTING VENTILATOR
0014 =          BEEP BIT 14H ;AUDIO ON/OFF
0006 =          SIL BIT 6H  ;TWO MIN SILENCE
0007 =          SPON_BR BIT 7H ;PATIENT BREATH
0008 =          HOLD BIT 8H  ;NEB OFF
0009 =          SEE_PIP BIT 09H ;DISPLAY PIP
000B =          DIV22 BIT 0BH  ;TIMER
000C =          ALM BIT 0CH ;AUDIO ALM SET
000D =          OFF_ALM BIT 0DH :BLINK_BEEP
000A =          ALM_TST BIT 0AH :SET DURING TEST
000E =          DIV24 BIT 0EH  ;START DELAY
000F =          FLOW BIT 0FH ;NEB FLOW ON
0010 =          SEE_TEMP BIT 10H
0011 =          SEE_LD BIT 11H
0012 =          DEL1 BIT 12H
0013 =          DEL_4TENTHS BIT 13H  :TIMER
0015 =          INSP BIT 15H ;INSP TIME
0016 =          CLOG1 BIT 16H :COUNT FLT LD SAMP
0017 =          CLOG2 BIT 17H
001C =          L14 BIT 1CH :LO BAT         LED1
001D =          L15 BIT 1DH ;FILTER CHANGE
001E =          L16 BIT 1EH :WAIT 5 CYCLES
001F =          L17 BIT 1FH ;LO FLOW
0034 =          L24 BIT 34H ;NO FLOW        LED2
0035 =          L25 BIT 35H :NEB HOLD
0036 =          L26 BIT 36H :FILT CLOG
```

```
0037 =          L27 BIT 37H :CONT FLOW
002C =          L34 BIT 2CH :HI PRESS       LED3
0020 =          L35 BIT 20H :HI TEMP
0038 =          DIV23 BIT 38H :TIMER
0039 =          CLK BIT 39H   ;TIMER 0.2S
003A =          HEAT BIT 3AH  :HEATER ON
003B =          TEMP BIT 3BH
0025            ENDS
                ;
0000            CSEG
                ; MACRO DEFINITIONS
                ;
                ANALOG MACRO SAVE      :ANALOG-DIGITAL CONVERSION
                NOP  ;DELAY TIME FOR MUX
                NOP
                NOP
                NOP
                NOP
                CLR P2.3      ;START CONVERSION
                NOP      ;ALLOW CONV. TIME 5 MICROSEC
                NOP
                NOP
                MOV SAVE,P1      ;SAVE DIGITAL OUTPUT
                SETB P2.3
                ENDM
                ;
                ;
                RUNNING_AVG MACRO LODATA,N.INSIG,AVG
                ;CALCULATES RUNNING AVERAGE OF N BYTES IN DATA MEMORY
                ;WITH A LOW ADDRESS OF LODATA. INPUT SIGNAL IS LOCATED
                :AT INSIG. AVERAGE OUTPUT IS AT AVG.
                PUSH PSW
                PUSH ACC
                PUSH B
                CLR PSW.3  ;BANKO
                CLR PSW.4
                MOV A,#LODATA    ;SET RO
                ADD A,#N
                DEC A
                MOV RO,A
                NEXT1:
                DEC RO
                MOV A,@RO     :SHIFT UP
                INC RO
                MOV @RO,A
                DEC RO
                CJNE RO,#LODATA,NEXT1  :LODATA ADDRESS
                MOV A,INSIG     ;MOV NEW DATA TO LODATA
                MOV B,#N
                DIV AB
                MOV @RO,A
                MOV A,#LODATA  :ADD TO CALC AVG
                ADD A,#N
                DEC A
                MOV TEMP_STORE,A
```

```
MOV A,@R0
XCH A,R0
NEXT2:
XCH A,R0
INC R0
ADD A,@R0
XCH A,R0
CJNE A,TEMP_STORE,NEXT2
XCH A,R0
MOV AVG,A
POP B
POP ACC
POP PSW
ENDM

FIFO MACRO NEW_IN,N1,NEW_DATA
;REGISTER STORES SUCCESSIVE DATA FIFO
;FROM NEW_DATA SOURCE INTO REGISTER ADDRESS
:NEW_IN. N1 IS THE NUMBER OF DATA STORED.
CLR PSW.3    :BANK0
CLR PSW.4
MOV A,#NEW_IN   ;SET R0
ADD A,#N1
DEC A
MOV R0,A
NEXT3:
DEC R0
MOV A,@R0   ;SHIFT UP
INC R0
MOV @R0,A
DEC R0
CJNE R0,#NEW_IN,NEXT3   ;NEW_IN ADDR
MOV NEW_IN,NEW_DATA
ENDM
BINARY_BCD MACRO HUN,TEN,ONE
;CONVERTS BYTE LOCATED IN ACC TO  DECIMAL
;AND STORES RESULT IN HUN, TEN AND ONE.
MOV HUN,#0   :CLEAR REGISTERS
MOV TEN,#0
MOV ONE,#0
CALC_HUN:   ;:SUBTRACT 100
MOV B,A
NEXTSUB1:
CLR C
SUBB A,#100
JC CALC_TEN
INC HUN
MOV B,A     :SAVE
SJMP NEXTSUB1
CALC_TEN:   ;:SUBTRACT 10
MOV A,B
NEXTSUB2:
CLR C
SUBB A,#10
JC CALC_ONE
```

```
                INC TEN
                MOV B,A
                SJMP NEXTSUB2
                CALC_ONE:
                MOV ONE,B
                MOV A,HUN
                JNZ BCD_OUT
                MOV HUN,#0FH   ;BLANK
                MOV A,TEN
                JNZ BCD_OUT
                MOV TEN,#0FH   ;BLANK
                BCD_OUT:
                ENDM
                :
                ;
1000            ORG 1000H
                ;%S
                BEGIN:
1000 0130       AJMP INITIALIZE
1003            ORG 1003H :MANUAL SWITCH INT..INT0
1003 0219CC     LJMP MAN_SW
100B            ORG 100BH ;TIMER 0 INT.,TF0
100B 61F3       AJMP TIM_SAMP
1013            ORG 1013H :LOW BATTERY INT.,INT1
1013 C28B       CLR IE1
1015 D21C       SETB L14
1017 852399     MOV SBUF,LED1
101A D125       ACALL TRANS_DEL
101C 32         RETI
                ;
1030            ORG 1030H
                INITIALIZE:   ;:SET REGISTERS
1030 D212       SETB DEL1
                INIT1:
1032 538700     ANL PCON,#00H ; SMOD = 0
1035 758920     MOV TMOD,#00100000B ; TIME 1 MODE 2, TIME 0 MODE 0
1038 759850     MOV SCON,#01010000B ; SERIAL PORT MODE 1
103B 758C70     MOV TH0,#70H   ;SET TIMER
103E 758DFD     MOV TH1,#0FDH ; BAUD RATE 9600
1041 75A078     MOV P2,#78H   ; OUTPUTS OFF
1044 75A887     MOV IE,#87H   ;ENABLE EX1,ET0,EX0
1047 75B802     MOV IP,#02H   :FIRST PRIORITY TIMER 0
104A 758850     MOV TCON,#50H :TIMERS ACTIVE, IT1 & IT0
                              ;LOW LEVEL TRIGGGER
104D 758000     MOV P0,#00H
1050 758130     MOV SP,#30H ; STACK ADDRESS
1053 752000     MOV 20H,#00H ;CLEAR BITS
1056 752100     MOV 21H,#00H
1059 752200     MOV 22H,#00H
105C 752700     MOV 27H,#00H
105F D2D3       SETB PSW.3      ;BANK1
1061 7B05       MOV RDIV5,#5    ;R3
1063 7A0A       MOV RDIV10,#10 :R2
1065 7E78       MOV RSIL_TIM,#120  :R6,DEL 2 MIN (3CH)
1067 7F78       MOV RHOLD_TIM,#120 :R7
```

```
1069 7D32      MOV ROFF_TIM,#NOFLOTIM ;R5, CLEAR REGISTER
106B 7C0B      MOV RON_TIM,#FLOTIM ;R4
106D 7900      MOV RCHG_TIM,#0 ;R1
106F 751532    MOV OFFTIMER,#NOFLOTIM
1072 75140B    MOV ONTIMER,#FLOTIM
1075 755000    MOV POSSUM,#0
1078 755100    MOV NEGSUM,#0
107B 751128    MOV TEMP_SET,#40  :DEFAULT
107E 752C00    MOV FLTLD_HUN,#00H
1081 752D01    MOV FLTLD_TEN,#01H
1084 752E02    MOV FLTLD_ONE,#02H
1087 751DFF    MOV DIVIDE1,#0FFH   ;TRANS DEL
108A 751E04    MOV DIVIDE2,#04H
108D D202      SETB DIV21  ;TIMER
108F D20B      SETB DIV22
1091 751B45    MOV VENT_HI,#45H  ;THRESH = 2.7V/2 = 1.35V
1094 751A3B    MOV VENT_LO,#3BH  ;THRESH = 2.3V/2 = 1.15V
1097 751928    MOV SET_CHGTIM,#40 ;CASE8 GIVES 60
109A C2D3      CLR PSW.3        ;BANK0
109C 7E13      MOV RVENT,#13H   ;R6,VENT #
109E 8E99      MOV SBUF,RVENT
10A0 D125      ACALL TRANS_DEL
10A2 752344    MOV LED1,#44H    ;WAIT LED ON
10A5 852399    MOV SBUF,LED1
10A8 D125      ACALL TRANS_DEL
10AA 752605    MOV LED2,#05H
10AD 852699    MOV SBUF,LED2
10B0 D125      ACALL TRANS_DEL
10B2 752506    MOV LED3,#06H
10B5 852599    MOV SBUF,LED3
10B8 D125      ACALL TRANS_DEL
10BA 752840    MOV CHG_VOL,#40H  :CASE8 GIVES 600ML
10BD 301212    JNB DEL1,CONT5
10C0 C212      CLR DEL1
10C2 C20B      CLR DIV22
10C4 C20E      CLR DIV24
10C6 200E02    DELAY1: JB DIV24,DELAY2
10C9 80FB      SJMP DELAY1
10CB 300E02    DELAY2: JNB DIV24,END_DEL
10CE 80FB      SJMP DELAY2
10D0 0132      END_DEL: AJMP INIT1
10D2 121765    CONT5: LCALL CASE81
10D5 00        NOP
10D6 00        NOP
10D7 00        NOP
               ;
               MAIN_LOOP:  ::INSP/EXP CYCLE
10D8 12156E    LCALL SERVICE
10DB 200C51    JB ALM.ALARM
10DE C2D3      CLR PSW.3 :BANK0
10E0 C2D4      CLR PSW.4
10E2 E51B      MOV A,VENT_HI ;WAIT FOR SOI
10E4 C3        CLR C
10E5 9560      SUBB A,VENT_AVG :R1
10E7 50EF      JNC MAIN_LOOP  ;?NOT INSP
```

```
10E9 D215       SETB INSP

                EOI: ;:WAIT FOR EOI
10EB 12156E     LCALL SERVICE
10EE 200C3E     JB ALM.ALARM
10F1 E51A       MOV A,VENT_LO
10F3 C3         CLR C
10F4 956C       SUBB A,VENT_AVG ;R1
10F6 40F3       JC EOI    ;?NOT EOI
10F8 C215       CLR INSP

10FA D2D3       SETB PSW.3  ;BANK1
10FC C2D4       CLR PSW.4
10FE 7900       MOV RCHG_TIM,#00H ;R1
                CHK_EXH:       ;:FIND AWP PEAK & DROP
1100 E54D       MOV A,AWP_MAX
1102 C3         CLR C
1103 954C       SUBB A,AWP_AVG
1105 401B       JC DELAY5      ;?AWP MAX > AWP AVG
                CHK_AWP:      ;:CHK AWP DROP
1107 F5F0       MOV B,A    ;SAVE
1109 E54D       MOV A,AWP_MAX
110B C3         CLR C
110C 9563       SUBB A,PEEP_AVG ;AWP MAX - PEEP
110E 4007       JC SET_EXH  ;AWP<PEEP
1110 84         DIV AB
1111 9405       SUBB A,#5
1113 4002       JC SET_EXH  ;?DROP 20%
1115 2122       AJMP DELAY5
1117 D201       SET_EXH:  SETB EXH
1119 854D4E     MOV PIP_STORE,AWP_MAX  ;NEW PIP
111C 754D00     MOV AWP_MAX,#0  ;RESET
111F 2150       AJMP CHK_PEAK

1121 00         NOP
                DELAY5:  ;:WAIT 0.5S
1122 D2D3       SETB PSW.3  ;BANK1
1124 C2D4       CLR PSW.4
1126 7432       MOV A,#50
1128 C3         CLR C
1129 99         SUBB A,RCHG_TIM
112A 50D4       JNC CHK_EXH ;?NOT 0.5S
112C 00         NOP
112D 00         NOP
112E 00         NOP

                ALARM:
112F D20C       SETB ALM
1131 43A070     ORL P2,#01110000B  ;OFF VALVES
1134 200605     CHK_SIL: JB SIL.CONT
1137 200802     JB HOLD,CONT
113A D2A7       SETB P2.7 ;BUZZER ON
113C D200       CONT: SETB WAIT
113E D21E       SETB L16  ;WAIT
1140 852299     MOV SBUF,LED1
```

```
 1143 D125      ACALL TRANS_DEL
 1145 12156E    LCALL SERVICE
 1148 200CE9    JB ALM.CHK_SIL
 114B 752F00    MOV THREE_CYCLE,#0
 114E 01D8      AJMP MAIN_LOOP

                CHK_PEAK:    ;:PRESS LIMIT 120 CM
 1150 20000A    JB WAIT,CALC_PIP
 1153 E54E      MOV A,PIP_STORE
 1155 C3        CLR C
 1156 9563      SUBB A,PEEP_AVG
 1158 C3        CLR C
 1159 94E0      SUBB A,#PIP_THRESH
 115B 5046      JNC HIPRESS
                CALC_PIP:
 115D           RUNNING_AVG PIP_LO,3,PIP_STORE,PIP_AVG
+               ;CALCULATES RUNNING AVERAGE OF 3 BYTES IN DATA MEMORY
+               ;WITH A LOW ADDRESS OF LODATA. INPUT SIGNAL IS LOCATED
+               ;AT INSIG. AVERAGE OUTPUT IS AT AVG.
+115D C0D0      PUSH PSW
+115F C0E0      PUSH ACC
+1161 C0F0      PUSH B
+1163 C2D3      CLR PSW.3   :BANK0
+1165 C2D4      CLR PSW.4
+1167 7458      MOV A,#PIP_LO    :SET R0
+1169 2403      ADD A,#3
+116B 14        DEC A
+116C F8        MOV R0,A
+               NEXT10001:
+116D 18        DEC R0
+116E E6        MOV A,@R0     ;SHIFT UP
+116F 08        INC R0
+1170 F6        MOV @R0,A
+1171 18        DEC R0
+1172 B858F8    CJNE R0,#PIP_LO,NEXT10001   :LODATA ADDRESS
+1175 E54E      MOV A,PIP_STORE     ;MOV NEW DATA TO PIP_LO
+1177 75F003    MOV B,#3
+117A 84        DIV AB
+117B F6        MOV @R0,A
+117C 7458      MOV A,#PIP_LO   ;ADD TO CALC PIP_AVG
+117E 2403      ADD A,#3
+1180 14        DEC A
+1181 F51C      MOV TEMP_STORE,A
+1183 E6        MOV A,@R0
+1184 C8        XCH A,R0
+               NEXT20001:
+1185 C8        XCH A,R0
+1186 08        INC R0
+1187 26        ADD A,@R0
+1188 C8        XCH A,R0
+1189 B51CF9    CJNE A,TEMP_STORE,NEXT20001
+118C C8        XCH A,R0
+118D F55B      MOV PIP_AVG,A
+118F D0F0      POP B
+1191 D0E0      POP ACC
```

```
+1193 D0D0      POP PSW
 1195 00        NOP
 1196 00        NOP
 1197 00        NOP
 1198 200013    JB WAIT.STRT_EXH
 119B 200810    JB HOLD,STRT_EXH
 119E 53A08F    ANL P2,#10001111B    ;ON VALVES
 11A1 21AE      AJMP STRT_EXH
                HIPRESS:
 11A3 D22C      SETB L34  ;HI PRESS
 11A5 852599    MOV SBUF,LED3
 11A8 D125      ACALL TRANS_DEL
 11AA 00        NOP
 11AB 212F      ALARM1: AJMP ALARM
 11AD 00        NOP

                STRT_EXH:
 11AE D2D3      SETB PSW.3           :BANK1
 11B0 C2D4      CLR PSW.4
 11B2 7900      MOV RCHG_TIM,#00H    :R1,RST CHARGE TIME

                CHARGE:
 11B4 C2D3      CLR PSW.3              :BANK 0
 11B6 12156E    LCALL SERVICE
 11B9 200CEF    JB ALM,ALARM1
 11BC E51B      MOV A,VENT_HI     :VENTILATOR INSPIRATION?
 11BE C3        CLR C
 11BF 956C      SUBB A,VENT_AVG
 11C1 5023      JNC CHK_CHGTIM      ;?NO VENT INSP1
 11C3 43A070    ORL P2,#01110000B   ;OFF VALVES

 11C6 D2D3      CHK_VOL: SETB PSW.3  ;BANK1
 11C8 C2D4      CLR PSW.4
 11CA 200016    JB WAIT,CHK_WAIT1
 11CD E519      MOV A,SET_CHGTIM
 11CF C3        CLR C
 11D0 99        SUBB A,RCHG_TIM  ;R1
 11D1 4010      JC CHK_WAIT1  ;:VOL>SET
 11D3 F5F0      MOV B.A
 11D5 E519      MOV A,SET_CHGTIM
 11D7 84        DIV AB
 11D8 940A      SUBB A,#10
 11DA 5007      JNC CHK_WAIT1
 11DC D21F      SETB L17  :LO FLOW LED
 11DE 852399    MOV SBUF.LED1
 11E1 D125      ACALL TRANS_DEL
 11E3 6108      CHK_WAIT1:  AJMP CHK_WAIT
 11E5 00        NOP
                CHK_CHGTIM:
 11E6 E519      MOV A,SET_CHGTIM  ;SET VOLUME REACHED?
 11E8 D2D3      SETB PSW.3        :BANK1
 11EA C3        CLR C
 11EB 99        SUBB A,RCHG_TIM :R1
 11EC 50C6      JNC CHARGE  :?VOL < SET VOL
 11EE 43A070    ORL P2,#01110000B     :OFF VALVES
```

```
11F1 20004B     JB WAIT,CHK_EOEXH1

11F4 101629     JBC CLOG1,FIRST_SAMP :MEAS FLT LD SAMP
11F7 301728     JNB CLOG2,FLT_LD
11FA C217       CLR CLOG2  :SECOND SAMPLE
11FC E544       MOV A,FLTFLO_AVG
11FE 2545       ADD A,CLOG_LO
1200 F546       MOV CLOG_HI,A  ;UPPER LIM FILT CLOG
1202 C3         CLR C
1203 13         RRC A  ;DIV BY 2
1204 F545       MOV CLOG_LO,A  :LOWER LIM FILT CLOG
1206 C3         CLR C
1207 13         RRC A  ;HALF CLOG LO
1208 F5F0       MOV B,A  ;SAVE
120A 2545       ADD A,CLOG_LO
120C F556       MOV FLTLD50,A  ;STORE 50% LEVEL
120E E5F0       MOV A,B
1210 C3         CLR C
1211 13         RRC A  ;ONE FOURTH CLOG LO
1212 F5F0       MOV B,A  ;SAVE
1214 2545       ADD A,CLOG_LO
1216 F555       MOV FLTLD25,A  ;STORE 25% LEVEL
1218 E5F0       MOV A,B
121A 2556       ADD A,FLTLD50
121C F557       MOV FLTLD75,A  :STORE 75% LEVEL
121E 4142       AJMP CHK_DPTHRESH

                FIRST_SAMP:   ;:FIRST FLT LD SAMP
1220 854445     MOV CLOG_LO,FLTFLO_AVG  ;SAVE
1223 4142       AJMP CHK_DPTHRESH

                FLT_LD: ;:SAVE FILT LOAD %
1225 E544       MOV A,FLTFLO_AVG
1227 C3         CLR C
1228 9546       SUBB A,CLOG_HI
122A 402F       JC TEST75
122C D236       SETB L26  ;FILTER CLOG LED
122E 852399     MOV SBUF,LED1
1231 D125       ACALL TRANS_DEL
1233 752C10     MOV FLTLD_HUN,#10H  ;SET FILTER LOAD 100%
1236 752D01     MOV FLTLD_TEN,#01H
1239 752E02     MOV FLTLD_ONE,#02H
123C 212F       AJMP ALARM

123E 00         NOP
123F 41A7       CHK_EOEXH1: AJMP CHK_EOEXH
1241 00         NOP

                CHK_DPTHRESH:
1242 E544       MOV A,FLTFLO_AVG
1244 C3         CLR C
1245 948D       SUBB A,#FILTDP_THRESH
1247 40F6       JC CHK_EOEXH1  ;BELOW THRESH
1249 D236       SETB L26  :FILT CLOG LED
124B 852699     MOV SBUF,LED2
```

```
124E D125       ACALL TRANS_DEL
1250 752C10     MOV FLTLD_HUN.#10H
1253 752D01     MOV FLTLD_TEN.#01H
1256 752E02     MOV FLTLD_ONE.#02H
1259 212F       AJMP ALARM

                TEST75:    ;:TEST 75% CLOG
125B E544       MOV A.FLTFLO_AVG
125D C3         CLR C
125E 9557       SUBB A.FLTLD75
1260 4012       JC TEST50
1262 D21D       SETB L15    ;FILTER CHANGE LED
1264 852399     MOV SBUF,LED1
1267 D125       ACALL TRANS_DEL
1269 752CF0     MOV FLTLD_HUN,#0F0H   ;BLANK
126C 752D71     MOV FLTLD_TEN,#71H
126F 752E52     MOV FLTLD_ONE,#52H
1272 4142       AJMP CHK_DPTHRESH
                TEST50:   ;:TEST 50% CLOG
1274 E544       MOV A,FLTFLO_AVG
1276 C3         CLR C
1277 9556       SUBB A,FLTLD50
1279 400B       JC TEST25
127B 752CF0     MOV FLTLD_HUN,#0F0H
127E 752D51     MOV FLTLD_TEN,#51H
1281 752E02     MOV FLTLD_ONE,#02H
1284 4142       AJMP CHK_DPTHRESH
                TEST25:    ;:TEST 25% CLOG
1286 E544       MOV A,FLTFLO_AVG
1288 C3         CLR C
1289 9555       SUBB A,FLTLD25
128B 4008       JC TEST0
128D 752CF0     MOV FLTLD_HUN,#0F0H
1290 752D21     MOV FLTLD_TEN,#21H
1293 752E52     MOV FLTLD_ONE,#52H
1296 4142       AJMP CHK_DPTHRESH
                TEST0:
1298 752CF0     MOV FLTLD_HUN,#0F0H
129B 752DF1     MOV FLTLD_TEN,#0F1H
129E 752E02     MOV FLTLD_ONE,#02H
12A1 4142       AJMP CHK_DPTHRESH
12A3 00         NOP

12A4 212F       ALARM2: AJMP ALARM
12A6 00         NOP
                CHK_EOEXH:
12A7 12156E     LCALL SERVICE
12AA 200CF7     JB ALM,ALARM2
12AD C2D3       CLR PSW.3   ;BANK0
12AF C2D4       CLR PSW.4
12B1 E518       MOV A,VENT_HI
12B3 956C       SUBB A,VENT_AVG   :R1
12B5 503F       JNC PAT_INSP
12B7 C201       CLR EXH   :END OF EXHALATION
12B9            RUNNING_AVG PEEP_LO.3.AWP_AVG.PEEP_AVG
```

```
+                    ;CALCULATES RUNNING AVERAGE OF 3 BYTES IN DATA MEMORY
+                    ;WITH A LOW ADDRESS OF LODATA. INPUT SIGNAL IS LOCATED
+                    ;AT INSIG. AVERAGE OUTPUT IS AT AVG.
+12B9 C0D0           PUSH PSW
+12BB C0E0           PUSH ACC
+12BD C0F0           PUSH B
+12BF C2D3           CLR PSW.3   ;BANK0
+12C1 C2D4           CLR PSW.4
+12C3 7460           MOV A,#PEEP_LO    ;SET R0
+12C5 2403           ADD A,#3
+12C7 14             DEC A
+12C8 F8             MOV R0,A
+                    NEXT10002:
+12C9 18             DEC R0
+12CA E6             MOV A,@R0      ;SHIFT UP
+12CB 08             INC R0
+12CC F6             MOV @R0,A
+12CD 18             DEC R0
+12CE B860F8         CJNE R0,#PEEP_LO,NEXT10002  ;LODATA ADDRESS
+12D1 E54C           MOV A,AWP_AVG      :MOV NEW DATA TO PEEP_LO
+12D3 75F003         MOV B,#3
+12D6 84             DIV AB
+12D7 F6             MOV @R0,A
+12D8 7460           MOV A,#PEEP_LO   ;ADD TO CALC PEEP_AVG
+12DA 2403           ADD A,#3
+12DC 14             DEC A
+12DD F51C           MOV TEMP_STORE,A
+12DF E6             MOV A,@R0
+12E0 C8             XCH A,R0
+                    NEXT20002:
+12E1 C8             XCH A,R0
+12E2 08             INC R0
+12E3 26             ADD A,@R0
+12E4 C8             XCH A,R0
+12E5 B51CF9         CJNE A,TEMP_STORE,NEXT20002
+12E8 C8             XCH A,R0
+12E9 F563           MOV PEEP_AVG,A
+12EB D0F0           POP B
+12ED D0E0           POP ACC
+12EF D0D0           POP PSW
 12F1 00             NOP
 12F2 00             NOP
 12F3 00             NOP
 12F4 6108           AJMP CHK_WAIT


                     PAT_INSP:
 12F6 E563           MOV A,PEEP_AVG
 12F8 C3             CLR C
 12F9 954C           SUBB A,AWP_AVG   :PEEP - AWP
 12FB 40AA           JC CHK_EOEXH     :AWP > PEEP
 12FD 9408           SUBB A,#PATINSP_THRESH
 12FF 40A6           JC CHK_EOEXH    :?NO PAT INSP
 1301 C201           CLR EXH
 1303 D207           SETB SPON_BR
```

```
1305 00          NOP
1306 00          NOP
1307 00          NOP

                 CHK_WAIT:  ::CHECK 3 CYC WAIT
1308 30002E      JNB WAIT.GO_ON
130B 20042B      JB VEN_SEL,GO_ON
130E 200328      JB VOL_CHG.GO_ON
1311 200825      JB HOLD.GO_ON
1314 7402        MOV A.#2
1316 C3          CLR C
1317 952F        SUBB A,THREE_CYCLE
1319 501B        JNC INC3
131B C200        CLR WAIT
131D C21E        CLR L16
131F D216        SETB CLOG1
1321 D217        SETB CLOG2
1323 852399      MOV SBUF,LED1
1326 D125        ACALL TRANS_DEL
1328 7C00        MOV RON_TIM.#0  :RESET AFTER WAIT
132A 7D00        MOV ROFF_TIM,#0
132C 755000      MOV POSSUM,#0
132F 755100      MOV NEGSUM,#0
1332 6139        AJMP GO_ON
1334 00          NOP
1335 00          NOP

                 INC3:
1336 052F        INC THREE_CYCLE
1338 00          NOP
                 GO_ON: ;:START MAIN LOOP
1339 01D8        AJMP MAIN_LOOP
133B 00          NOP

133C 61EB        OUT1: AJMP OUT
133E 00          NOP

                 BLINK_BEEP:  ::ON/OFF DISPLAY & BUZZER
133F 1038FA      JBC DIV23.OUT1 ;PERIOD 0.4S
1342 D238        SETB DIV23
1344 200AF5      JB ALM_TST,OUT1
1347 C2D3        CLR PSW.3   ;BANK0
1349 C2D4        CLR PSW.4
134B 100D50      JBC OFF_ALM,TURN_OFF
                 TURN_ON:  ;:DISPLAY/ALM ON
134E D20D        SETB OFF_ALM
1350 301F05      JNB L17.CHK_LED21
1353 852399      MOV SBUF,LED1   ;RESTORE LED'S
1356 D125        ACALL TRANS_DEL
                 CHK_LED21:
1358 E526        MOV A.LED2
135A 54F0        ANL A.#0F0H
135C 6005        JZ CHK_LED31
135E 852699      MOV SBUF,LED2
1361 D125        ACALL TRANS_DEL
```

```
                    CHK_LED31:
1363 E525           MOV A,LED3
1365 54F0           ANL A,#0F0H
1367 6005           JZ CHK_VOL1
1369 852599         MOV SBUF,LED3
136C D125           ACALL TRANS_DEL
136E 30030F         CHK_VOL1:  JNB VOL_CHG.TST_VENTSEL1
1371 852899         MOV SBUF,CHG_VOL  ;SET HUNS
1374 D125           ACALL TRANS_DEL
1376 759901         MOV SBUF,#01H    ;SET TENS TO 0
1379 D125           ACALL TRANS_DEL
137B 759902         MOV SBUF,#02H    ;SET ONES TO 0
137E D125           ACALL TRANS_DEL
                    TST_VENTSEL1:
1380 300404         JNB VEN_SEL,TST_TEMP1
1383 8E99           MOV SBUF,RVENT
1385 D125           ACALL TRANS_DEL
                    TST_TEMP1:
1387 303B0A         JNB TEMP,TST_BEEP1
138A 851299         MOV SBUF,TEMP_DEC  ;TENS
138D D125           ACALL TRANS_DEL
138F 759902         MOV SBUF,#02H ;ONES
1392 D125           ACALL TRANS_DEL
                    TST_BEEP1:
1394 301454         JNB BEEP,OUT
1397 200651         JB SIL,OUT
139A D2A7           SETB P2.7    ;BUZZER ON
139C 61EB           AJMP OUT
                    TURN_OFF:    ::DISPLAY/ALM OFF
139E 301F08         JNB L17,CHK_LED22
13A1 E523           MOV A,LED1
13A3 547F           ANL A,#7FH    ;MASK LED'S
13A5 F599           MOV SBUF,A
13A7 D125           ACALL TRANS_DEL
                    CHK_LED22:
13A9 E526           MOV A,LED2
13AB 54F0           ANL A,#0F0H
13AD 6005           JZ CHK_LED32
13AF 759905         MOV SBUF,#05H
13B2 D125           ACALL TRANS_DEL
                    CHK_LED32:
13B4 E525           MOV A,LED3
13B6 54F0           ANL A,#0F0H
13B8 6005           JZ CHK_VOL2
13BA 759906         MOV SBUF,#06H
13BD D125           ACALL TRANS_DEL
13BF 30030F         CHK_VOL2: JNB VOL_CHG.TST_VENTSEL2
13C2 7599F0         MOV SBUF,#0F0H    ;OFF HUNS
13C5 D125           ACALL TRANS_DEL
13C7 7599F1         MOV SBUF,#0F1H    ;OFF TENS
13CA D125           ACALL TRANS_DEL
13CC 7599F2         MOV SBUF,#0F2H    ;OFF ONES
13CF D125           ACALL TRANS_DEL
                    TST_VENTSEL2:
13D1 300405         JNB VEN_SEL,TST_TEMP2
```

```
13D4 7599F3     MOV SBUF,#0F3H     :VENT SEL OFF
13D7 D125       ACALL TRANS_DEL
                TST_TEMP2:
13D9 303B0A     JNB TEMP.TST_BEEP2
13DC 7599F1     MOV SBUF,#0F1H   :OFF TENS
13DF D125       ACALL TRANS_DEL
13E1 7599F2     MOV SBUF,#0F2H   :OFF ONES
13E4 D125       ACALL TRANS_DEL

                TST_BEEP2:
13E6 301402     JNB BEEP.OUT
13E9 C2A7       CLR P2.7    :AUDIO OFF
                OUT:
13EB 758C70     MOV TH0,#70H   :RST TIMER0
13EE D2A9       SETB ET0
13F0 D28C       SETB TR0
13F2 22         RET


                TIM_SAMP:      ::TIMER 0 INTERRUPT
13F3 C0E0       PUSH ACC    ;SAVE SFR'S
13F5 C0F0       PUSH B
13F7 C0D0       PUSH PSW
13F9 758C70     MOV TH0,#70H   ;RESET TIMER
13FC D2D3       SETB PSW.3    ;SELECT REGISTER BANK 1
13FE C2D4       CLR PSW.4
1400 100204     JBC DIV21,CLEAR
1403 D202       SETB DIV21 ;FREQ 100HZ
1405 A167       AJMP RETURN
1407 09         CLEAR:  INC RCHG_TIM ;R1
1408 DA2B       DJNZ RDIV10,SAMPLE :R2
140A 7A0A       MOV RDIV10,#10 ;RESET RDIV10
140C 100B04     JBC DIV22,SET_CLK
140F D20B       SETB DIV22
1411 8135       AJMP SAMPLE

                SET_CLK:   ::SET .2S CLOCK
1413 D239       SETB CLK
1415 100E02     JBC DIV24,CONT6
1418 D20E       SETB DIV24
141A DB19       CONT6: DJNZ RDIV5,SAMPLE   ;R3
141C 7B05       MOV RDIV5,#5   ;FREQ 1 HZ

                ;SILENCE 2 MIN
141E 300608     JNB SIL,CHK_HOLD
1421 C2A7       CLR P2.7              ;BUZZER OFF
1423 DE04       DJNZ RSIL_TIM,CHK_HOLD     :?NOT 2 MIN
1425 7E78       MOV RSIL_TIM,#120 :R6. RESET 2 MIN
1427 C206       CLR SIL
                CHK_HOLD:   ::STOP NEB?
1429 300809     JNB HOLD,SAMPLE
142C DF07       DJNZ RHOLD_TIM,SAMPLE :R7
142E 7F78       MOV RHOLD_TIM,#120
1430 200602     JB SIL,SAMPLE
1433 D2A7       SETB P2.7    :ON BUZZER
```

```
                    SAMPLE:    ::READ VENT SIG
 1435 C2D3          CLR PSW.3    :BANK 0
 1437 C2D4          CLR PSW.4
 1439 53A0F8        ANL P2.#11111000B    :CLEAR MUX ADDRESS
 143C D2A3          SETB P2.3
 143E               ANALOG RVENT_SIG
+143E 00            NOP  ;DELAY TIME FOR MUX
+143F 00            NOP
+1440 00            NOP
+1441 00            NOP
+1442 00            NOP
+1443 C2A3          CLR P2.3      :START CONVERSION
+1445 00            NOP     ;ALLOW CONV. TIME 5 MICROSEC
+1446 00            NOP
+1447 00            NOP
+1448 A990          MOV RVENT_SIG,P1      ;SAVE DIGITAL OUTPUT
+144A D2A3          SETB P2.3
 144C 00            NOP
 144D               RUNNING_AVG VENT_LOW,4,RVENT_SIG,VENT_AVG
+                   ;CALCULATES RUNNING AVERAGE OF 4 BYTES IN DATA MEMORY
+                   ;WITH A LOW ADDRESS OF LODATA. INPUT SIGNAL IS LOCATED
+                   :AT INSIG. AVERAGE OUTPUT IS AT AVG.
+144D C0D0          PUSH PSW
+144F C0E0          PUSH ACC
+1451 C0F0          PUSH B
+1453 C2D3          CLR PSW.3   :BANK0
+1455 C2D4          CLR PSW.4
+1457 7468          MOV A,#VENT_LOW    :SET R0
+1459 2404          ADD A,#4
+145B 14            DEC A
+145C F8            MOV R0,A
+                   NEXT10004:
+145D 18            DEC R0
+145E E6            MOV A,@R0     :SHIFT UP
+145F 08            INC R0
+1460 F6            MOV @R0,A
+1461 18            DEC R0
+1462 B868F8        CJNE R0,#VENT_LOW,NEXT10004  :LODATA ADDRESS
+1465 E9            MOV A,RVENT_SIG     ;MOV NEW DATA TO VENT_LOW
+1466 75F004        MOV B,#4
+1469 84            DIV AB
+146A F6            MOV @R0,A
+146B 7468          MOV A,#VENT_LOW   :ADD TO CALC VENT_AVG
+146D 2404          ADD A,#4
+146F 14            DEC A
+1470 F51C          MOV TEMP_STORE,A
+1472 E6            MOV A,@R0
+1473 C8            XCH A,R0
+                   NEXT20004:
+1474 C8            XCH A,R0
+1475 08            INC R0
+1476 26            ADD A,@R0
+1477 C8            XCH A,R0
+1478 B51CF9        CJNE A,TEMP_STORE,NEXT20004
```

```
+147B C8          XCH A.R0
+147C F56C        MOV VENT_AVG.A
+147E D0F0        POP B
+1480 D0E0        POP ACC
+1482 D0D0        POP PSW
 1484 00          NOP
 1485 05A0        INC P2
 1487             ANALOG RFLT_FLO
+1487 00          NOP  :DELAY TIME FOR MUX
+1488 00          NOP
+1489 00          NOP
+148A 00          NOP
+148B 00          NOP
+148C C2A3        CLR P2.3      ;START CONVERSION
+148E 00          NOP     :ALLOW CONV. TIME 5 MICROSEC
+148F 00          NOP
+1490 00          NOP
+1491 AA90        MOV RFLT_FLO,P1      :SAVE DIGITAL OUTPUT
+1493 D2A3        SETB P2.3
 1495 00          NOP
 1496             RUNNING_AVG FLTFLO_LO,4,RFLT_FLO,FLTFLO_AVG
+                 :CALCULATES RUNNING AVERAGE OF 4 BYTES IN DATA MEMORY
+                 ;WITH A LOW ADDRESS OF LODATA. INPUT SIGNAL IS LOCATED
+                 :AT INSIG. AVERAGE OUTPUT IS AT AVG.
+1496 C0D0        PUSH PSW
+1498 C0E0        PUSH ACC
+149A C0F0        PUSH B
+149C C2D3        CLR PSW.3  ;BANK0
+149E C2D4        CLR PSW.4
+14A0 7440        MOV A,#FLTFLO_LO    :SET R0
+14A2 2404        ADD A,#4
+14A4 14          DEC A
+14A5 F8          MOV R0,A
+                 NEXT10006:
+14A6 18          DEC R0
+14A7 E6          MOV A,@R0     :SHIFT UP
+14A8 08          INC R0
+14A9 F6          MOV @R0,A
+14AA 18          DEC R0
+14AB B840F8      CJNE R0,#FLTFLO_LO,NEXT10006  ;LODATA ADDRESS
+14AE EA          MOV A,RFLT_FLO     :MOV NEW DATA TO FLTFLO_LO
+14AF 75F004      MOV B,#4
+14B2 84          DIV AB
+14B3 F6          MOV @R0,A
+14B4 7440        MOV A,#FLTFLO_LO  ;ADD TO CALC FLTFLO_AVG
+14B6 2404        ADD A,#4
+14B8 14          DEC A
+14B9 F51C        MOV TEMP_STORE,A
+14BB E6          MOV A,@R0
+14BC C8          XCH A,R0
+                 NEXT20006:
+14BD C8          XCH A,R0
+14BE 08          INC R0
+14BF 26          ADD A,@R0
+14C0 C8          XCH A,R0
```

```
+14C1 B51CF9    CJNE A,TEMP_STORE,NEXT20006
+14C4 C8        XCH A,R0
+14C5 F544      MOV FLTFLO_AVG,A
+14C7 D0F0      POP B
+14C9 D0E0      POP ACC
+14CB D0D0      POP PSW
 14CD 00        NOP
 14CE 05A0      INC P2
 14D0           ANALOG RAW_PRESS
+14D0 00        NOP  ;DELAY TIME FOR MUX
+14D1 00        NOP
+14D2 00        NOP
+14D3 00        NOP
+14D4 00        NOP
+14D5 C2A3      CLR P2.3      ;START CONVERSION
+14D7 00        NOP      ;ALLOW CONV. TIME 5 MICROSEC
+14D8 00        NOP
+14D9 00        NOP
+14DA A890      MOV RAW_PRESS,P1      ;SAVE DIGITAL OUTPUT
+14DC D2A3      SETB P2.3
 14DE 00        NOP
 14DF           RUNNING_AVG AWP_LO,4,RAW_PRESS,AWP_AVG
+              ;CALCULATES RUNNING AVERAGE OF 4 BYTES IN DATA MEMORY
+              ;WITH A LOW ADDRESS OF LODATA. INPUT SIGNAL IS LOCATED
+              ;AT INSIG. AVERAGE OUTPUT IS AT AVG.
+14DF C0D0      PUSH PSW
+14E1 C0E0      PUSH ACC
+14E3 C0F0      PUSH B
+14E5 C2D3      CLR PSW.3  ;BANK0
+14E7 C2D4      CLR PSW.4
+14E9 7448      MOV A,#AWP_LO   :SET R0
+14EB 2404      ADD A,#4
+14ED 14        DEC A
+14EE F8        MOV R0,A
+              NEXT10008:
+14EF 18        DEC R0
+14F0 E6        MOV A,@R0     :SHIFT UP
+14F1 08        INC R0
+14F2 F6        MOV @R0,A
+14F3 18        DEC R0
+14F4 B848F8    CJNE R0,#AWP_LO,NEXT10008  ;LODATA ADDRESS
+14F7 E8        MOV A,RAW_PRESS     :MOV NEW DATA TO AWP_LO
+14F8 75F004    MOV B,#4
+14FB 84        DIV AB
+14FC F6        MOV @R0,A
+14FD 7448      MOV A,#AWP_LO  :ADD TO CALC AWP_AVG
+14FF 2404      ADD A,#4
+1501 14        DEC A
+1502 F51C      MOV TEMP_STORE,A
+1504 E6        MOV A,@R0
+1505 C8        XCH A,R0
+              NEXT20008:
+1506 C8        XCH A,R0
+1507 08        INC R0
+1508 26        ADD A,@R0
```

```
+1509 C8          XCH A,R0
+150A B51CF9      CJNE A,TEMP_STORE,NEXT20008
+150D C8          XCH A,R0
+150E F54C        MOV AWP_AVG,A
+1510 D0F0        POP B
+1512 D0E0        POP ACC
+1514 D0D0        POP PSW
 1516 00          NOP
 1517 30150A      JNB INSP,NEXT_SAMP
 151A E54D        MOV A,AWP_MAX
 151C C3          CLR C
 151D 954C        SUBB A,AWP_AVG
 151F 5003        JNC NEXT_SAMP
 1521 854C4D      MOV AWP_MAX,AWP_AVG
                  NEXT_SAMP:
 1524 00          NOP
 1525 05A0        INC P2
 1527             ANALOG RNEB_FLO
+1527 00          NOP  :DELAY TIME FOR MUX
+1528 00          NOP
+1529 00          NOP
+152A 00          NOP
+152B 00          NOP
+152C C2A3        CLR P2.3      ;START CONVERSION
+152E 00          NOP      ;ALLOW CONV. TIME 5 MICROSEC
+152F 00          NOP
+1530 00          NOP
+1531 AC90        MOV RNEB_FLO,P1      ;SAVE DIGITAL OUTPUT
+1533 D2A3        SETB P2.3
 1535 00          NOP
 1536 EC          MOV A,RNEB_FLO  :R4
 1537 C3          CLR C
 1538 9432        SUBB A,#50
 153A 400E        JC NEGFLO
 153C C3          CLR C  :DIV BY 4
 153D 13          RRC A
 153E C3          CLR C
 153F 13          RRC A
 1540 2550        ADD A,POSSUM ;SUN POS FLOW
 1542 F550        MOV POSSUM,A  ;SAVE
 1544 500F        JNC CONT1
 1546 D20F        SETB FLOW  ;OVERFLOW CONDITION
 1548 800B        SJMP CONT1
                  NEGFLO:  ::NEG FLOW
 154A 7432        MOV A,#50
 154C 9C          SUBB A,RNEB_FLO
 154D C3          CLR C  :DIV BY 4
 154E 13          RRC A
 154F C3          CLR C
 1550 13          RRC A
 1551 2551        ADD A,NEGSUM
 1553 F551        MOV NEGSUM,A  ;SAVE
                  CONT1:
 1555 00          NOP
 1556 05A0        INC P2
```

```
 1558                ANALOG RTEMP
+1558 00            NOP   :DELAY TIME FOR MUX
+1559 00            NOP
+155A 00            NOP
+155B 00            NOP
+155C 00            NOP
+155D C2A3          CLR P2.3      :START CONVERSION
+155F 00            NOP      :ALLOW CONV. TIME 5 MICROSEC
+1560 00            NOP
+1561 00            NOP
+1562 AD90          MOV RTEMP,P1      ;SAVE DIGITAL OUTPUT
+1564 D2A3          SETB P2.3
 1566 00            NOP
                    RETURN:   ;:RET FROM INT
 1567 D0D0          POP PSW
 1569 D0F0          POP B
 156B D0E0          POP ACC
 156D 32            RETI


                    SERVICE: ;:CHK FLOW,SER-REC,BLINK
 156E 103902        JBC CLK,TEMP_CONT
 1571 22            RET
 1572 00            NOP
                    TEMP_CONT:   ;:CONTROL HEATER
 1573 C2D3          CLR PSW.3   ;BANK0
 1575 C2D4          CLR PSW.4
 1577 ED            MOV A.RTEMP   ;R5
 1578 B4A00C        CJNE A,#TEMP_HI,NOT_EQ
                    HITEMP:   ;:OVER 80C
 157B C281          CLR P0.1   ;HEAT OFF
 157D C23A          CLR HEAT
 157F D220          SETB L35 ;HI TEMP LED
 1581 852599        MOV SBUF,LED3
 1584 D20C          SETB ALM
 1586 22            RET
 1587 5022          NOT_EQ: JNC HI_TEMP ;RTEMP>TEMP_HI
 1589 E511          MOV A.TEMP_SET
 158B B42804        CJNE A,#40,HEAT_CHK
 158E C281          CLR P0.1   ;HEAT OFF
 1590 A1B5          AJMP FLO_TST
                    HEAT_CHK: ::CHK HEAT BIT
 1592 203A0C        JB HEAT,SW_OFF
 1595 C3            CLR C
 1596 940A          SUBB A,#10   ;LOW LIMIT
 1598 9D            SUBB A,RTEMP ;R5
 1599 401A          JC FLO_TST   ;?LEAVE OFF?
 159B D281          SETB P0.1   :TURN ON
 159D D23A          SETB HEAT
 159F A1B5          AJMP FLO_TST
                    SW_OFF:
 15A1 240A          ADD A.#10   :UPPER LIMIT
 15A3 C3            CLR C
 15A4 9D            SUBB A.RTEMP
 15A5 500E          JNC FLO_TST   ;?LEAVE ON?
```

```
15A7 C281       CLR P0.1  :TURN OFF
15A9 C23A       CLR HEAT
                HI_TEMP:  ;:TEMP ALARM
15AB D220       SETB L35  :HI TEMP LED
15AD 852599     MOV SBUF,LED3
15B0 D125       ACALL TRANS_DEL
15B2 D20C       SETB ALM
15B4 22         RET
                FLO_TST:  ;:TEST NEB FLOW
15B5 200070     JB WAIT,CHK_SERPORT
15B8 D2D3       SETB PSW.3  ;BANK1
15BA C2D4       CLR PSW.4
15BC 100F1C     JBC FLOW,FLO
15BF E550       MOV A,POSSUM
15C1 C3         CLR C
15C2 9551       SUBB A,NEGSUM ;CALC SFLO
15C4 5004       JNC CONT2
15C6 DD17       DJNZ ROFF_TIM,CONT4
15C8 A1F7       AJMP NOFLO_ALM
                CONT2:
15CA F5F0       MOV B,A  ;SAVE SFLO=POS-NEG
15CC 948C       SUBB A,#NOFLO_TH  :SFLO-THRESH
15CE 5004       JNC CONT3
15D0 DD02       NOFLO:  DJNZ ROFF_TIM,CONT3
15D2 A1F7       AJMP NOFLO_ALM
                CONT3:
15D4 E5F0       MOV A,B  ;SFLO
15D6 C3         CLR C
15D7 942D       SUBB A,#FLO_TH  ;SFLO-THRESH
15D9 4004       JC CONT4  ;?SFLO<THRESH
15DB DC02       FLO: DJNZ RON_TIM,CONT4
15DD C10E       AJMP FLO_ALM
                CONT4:  ::CHECK TIME
15DF 755000     MOV POSSUM,#0  ;RESET FLOW SUM
15E2 755100     MOV NEGSUM,#0
15E5 D51405     DJNZ ONTIMER,CHK_OFFTIM
15E8 75140B     MOV ONTIMER,#FLOTIM
15EB 7C0B       MOV RON_TIM,#FLOTIM
                CHK_OFFTIM:
15ED D51538     DJNZ OFFTIMER,CHK_SERPORT
15F0 751532     MOV OFFTIMER,#NOFLOTIM
15F3 7D32       MOV ROFF_TIM,#NOFLOTIM
15F5 C128       AJMP CHK_SERPORT

                NOFLO_ALM:  :?NEB OFF > 10S
15F7 755000     MOV POSSUM,#0
15FA 755100     MOV NEGSUM,#0
15FD 751532     MOV OFFTIMER,#NOFLOTIM
1600 7D32       MOV ROFF_TIM,#NOFLOTIM
1602 D214       SETB BEEP
1604 D20C       SETB ALM
1606 D234       SETB L24   :NO FLOW LED
1608 852699     MOV SBUF,LED2
160B D125       ACALL TRANS_DEL
160D 22         RET
```

```
                FLO_ALM:    ;:NEE IN : 3.2S
160E 755000     MOV POSSUM,#0
1611 755100     MOV NEGSUM,#0
1614 75140B     MOV ONTIMER,#FLOTIM
1617 7C0B       MOV RON_TIM,#FLOTIM
1619 D20C       SETB ALM    ;FLAG
161B D237       SETB L27   ;CONT FLOW ALM
161D 852699     MOV SBUF,LED2
1620 D125       ACALL TRANS_DEL
1622 22         RET

1623 613F       BLINK_BEEP1:  AJMP BLINK_BEEP
                TRANS_DEL: ;:DELAY 2.25MS.OSCFREE-
1625 02198E     LJMP TRANS_DEL1

                CHK_SERPORT:  ;:NEW CHAR REC?
1628 3098F8     JNB RI,BLINK_BEEP1
162B C298       CLR RI
162D C2A9       CLR ETO  ;DISABLE TIMER 0 INT
162F C28C       CLR TR0  ;DISABLE TIMER 0
1631 E599       MOV A,SBUF    ;READ CODE RECEIVED
1633 C4         SWAP A
1634 23         RL A    ;MULTIPLY BY 2
1635 901639     MOV DPTR,#JUMP_TBLE1
1638 73         JMP @A+DPTR
1639 C17D       JUMP_TBLE1: AJMP CASE0    ;TEMP. SET
163B C1F4                   AJMP CASE1    ;NEG. FLO
163D E19F                   AJMP CASE2    ;SELF TEST
163F E19D                   AJMP CASE3    ;NO ACTION
1641 E126                   AJMP CASE4    ;VENT SEL
1643 E1A1                   AJMP CASE5    ;DISPLAY TEMP
1645 C15D                   AJMP CASE61   ;ALM SEL
1647 C161                   AJMP CASE71   ;NO ACTION
1649 E169                   AJMP CASE8    ;CHANGE VOL
164B C165                   AJMP CASE91   ;DISPLAY FLOW
164D C169                   AJMP CASEA1   ;ALM RESET
164F C16D                   AJMP CASEB1   ;NO ACTION
1651 C171                   AJMP CASEC1   ;ENTER
1653 C175                   AJMP CASED1   ;DISPLAY PIP
1655 C179                   AJMP CASEE1   ;ALM TEST
1657 C15A                   AJMP CASEF    ;NO ACTION
1659 00         NOP
                CASEF: ;NO ACTION
165A 613F       AJMP BLINK_BEEP
165C 00         NOP
165D 02186D     CASE61: LJMP CASE8
1660 00         NOP
1661 021867     CASE71: LJMP CASE7
1664 00         NOP
1665 021818     CASE91: LJMP CASE9
1668 00         NOP
1669 02187A     CASEA1: LJMP CASEA
166C 00         NOP
166D 02186A     CASEB1: LJMP CASEB
```

```
1670 00          NOP
1671 021843      CASEC1: LJMP CASEC
1674 00          NOP
1675 0218D7      CASED1: LJMP CASED
1678 00          NOP
1679 021955      CASEE1: LJMP CASEE
167C 00          NOP

                 CASE0:   ;:TEMP SET
167D 203B19      JB TEMP,NEW_TEMP
1680 D23B        SETB TEMP
1682 E511        MOV A,TEMP_SET
1684 B42820      CJNE A,#40,DISPLAY_TEMP
                 OFF_STATE: ;:LCD "- -"
1687 7599F0      MOV SBUF,#0F0H ;HUNS BL
168A D125        ACALL TRANS_DEL
168C 7599A1      MOV SBUF,#0A1H ;TENS "-
168F D125        ACALL TRANS_DEL
1691 7599A2      MOV SBUF,#0A2H ;ONES "-
1694 D125        ACALL TRANS_DEL
1696 613F        AJMP BLINK_BEEP
1698 00          NOP
                 NEW_TEMP: ;:NEXT SET TE
1699 E511        MOV A,TEMP_SET
169B B47805      CJNE A,#120,CALC_TEMP
169E 751128      MOV TEMP_SET,#40
16A1 C187        AJMP OFF_STATE

                 CALC_TEMP:
16A3 2414        ADD A,#20
16A5 F511        MOV TEMP_SET,A

                 DISPLAY_TEMP:
16A7 C3          CLR C
16A8 13          RRC A   ;DIV BY 2
16A9             BINARY_BCD DEC_HUN,DEC_
+                ;CONVERTS BYTE LOCATED
+                ;AND STORES RESULT IN DE
+16A9 752900     MOV DEC_HUN,#0   ;CLEAR
+16AC 752A00     MOV DEC_TEN,#0
+16AF 752B00     MOV DEC_ONE,#0
+                CALC_HUN0011:   ;:SUBTR
+16B2 F5F0       MOV B,A
+                NEXTSUB10011:
+16B4 C3         CLR C
+16B5 9464       SUBB A,#100
+16B7 4006       JC CALC_TEN0011
+16B9 0529       INC DEC_HUN
+16BB F5F0       MOV B,A     ;SAVE
+16BD 80F5       SJMP NEXTSUB10011
+                CALC_TEN0011:   ;:SUBTR
+16BF E5F0       MOV A,B
+                NEXTSUB20011:
+16C1 C3         CLR C
+16C2 940A       SUBB A,#10
```

```
+16C4 4006      JC CALC_ONE0011
+16C6 052A      INC DEC_TEN
+16C8 F5F0      MOV B,A
+16CA 80F5      SJMP NEXTSUB20011
+               CALC_ONE0011:
+16CC 85F02B    MOV DEC_ONE,B
+16CF E529      MOV A,DEC_HUN
+16D1 700A      JNZ BCD_OUT0011
+16D3 75290F    MOV DEC_HUN,#0FH  ;BLANK
+16D6 E52A      MOV A,DEC_TEN
+16D8 7003      JNZ BCD_OUT0011
+16DA 752A0F    MOV DEC_TEN,#0FH  ;BLANK
+               BCD_OUT0011:
 16DD 7599F0    MOV SBUF,#0F0H  ;HUN BLANK
 16E0 D125      ACALL TRANS_DEL
 16E2 E52A      MOV A,DEC_TEN
 16E4 C4        SWAP A
 16E5 4401      ORL A,#01H
 16E7 F512      MOV TEMP_DEC,A  ;SAVE TENS
 16E9 F599      MOV SBUF,A
 16EB D125      ACALL TRANS_DEL
 16ED 759902    MOV SBUF,#02H  ;ONES
 16F0 D125      ACALL TRANS_DEL
 16F2 613F      AJMP BLINK_BEEP

                CASE1:   ;:NEBULIZER HOLD
 16F4 D2D3      SETB PSW.3   :BANK1
 16F6 C2D4      CLR PSW.4
 16F8 100810    JBC HOLD,HOLD_OFF
 16FB D208      SETB HOLD    ;HOLD FLAG
 16FD D214      SETB BEEP
 16FF D235      SETB L25     ;NEB HOLD LED
 1701 852699    MOV SBUF,LED2
 1704 D125      ACALL TRANS_DEL
 1706 43A070    ORL P2,#01110000B   ;OFF VALVES
 1709 800B      SJMP HOLD_OUT
                HOLD_OFF:
 1708 C208      CLR HOLD    ;HOLD FLAG
 170D C214      CLR BEEP
 170F C235      CLR L25     ;OFF HOLD LED
 1711 852699    MOV SBUF,LED2
 1714 D125      ACALL TRANS_DEL
                HOLD_OUT:
 1716 7F78      MOV RHOLD_TIM,#120 ;R7 RESET
 1718 D200      SETB WAIT
 171A D21E      SETB L16 ;WAIT LED
 171C 852399    MOV SBUF,LED1
 171F D125      ACALL TRANS_DEL
 1721 752F00    MOV THREE_CYCLE,#0
 1724 613F      AJMP BLINK_BEEP

                CASE4:   ::SELECT VENT
 1726 43A070    ORL P2,#01110000B  :VALVES OFF
 1729 D204      SETB VEN_SEL
 172B D200      SETB WAIT
```

```
172D 752F00     MOV THREE_CYCLE.#0
1730 D21E       SETB L16  :WAIT
1732 852399     MOV SBUF.LED1
1735 D125       ACALL TRANS_DEL
1737 C2D3       CLR PSW.3    ;BANKO
1739 C2D4       CLR PSW.4
173B EE         MOV A.RVENT    ;R4, INC. VENT. NO.
173C 2410       ADD A,#10H
173E B44302     CJNE A,#43H,SEE_VENT
1741 7413       MOV A,#13H   ;RESET #1
                SEE_VENT:
1743 FE         MOV RVENT,A
1744 8E99       MOV SBUF,RVENT    ;DISPLAY NEW NUMBER
1746 D125       ACALL TRANS_DEL
1748 00         NOP
1749 EE         MOV A,RVENT    ;LOOK UP THRESHOLDS FOR VENTILATOR SELECTED
174A C4         SWAP A
174B 540F       ANL A,#0FH    ;CLEAR ADDRESS
174D 23         RL A    ;MULT. BY 2
174E F5F0       MOV B,A     ;SAVE
1750 F15D       ACALL VENT_TBLE
1752 F51B       MOV VENT_HI,A    ;STORE UPPER THRESH
1754 E5F0       MOV A,B
1756 14         DEC A
1757 F15D       ACALL VENT_TBLE
1759 F51A       MOV VENT_LO,A    ;STORE LOWER THRESH
175B 613F       AJMP BLINK_BEEP

175D 83         VENT_TBLE: MOVC A,@A+PC
175E 22         RET    ;THRESHOLDS
175F 3B 45 81   DB 3BH,45H,81H,86H,3BH,45H ;SERVO LO 2.3V, HI 2.7V
1762 86 3B 45
                ;PB7200 LO 5.05V, HI 5.25V,  HAM LO 2.3V, HI 2.7V

                CASE81: ::INITIALIZATION ENTRY
1765 C2A9       CLR ETO
1767 C28C       CLR TRO

                CASE8:    ;:CHANGE VOLUME
1769 D203       SETB VOL_CHG
176B E528       MOV A,CHG_VOL
176D C4         SWAP A
176E 23         RL A
176F F5F0       MOV B,A
1771 F18F       ACALL CHGVOL_TBLE
1773 F519       MOV SET_CHGTIM,A
1775 E5F0       MOV A,B
1777 14         DEC A
1778 F18F       ACALL CHGVOL_TBLE
177A F528       MOV CHG_VOL,A
177C 852899     MOV SBUF,CHG_VOL
177F D125       ACALL TRANS_DEL
1781 D200       SETB WAIT
1783 752F00     MOV THREE_CYCLE,#0
1786 D21E       SETB L16
```

```
1788 852399     MOV SBUF,LED1
1788 D125       ACALL TRANS_DEL
178D 613F       AJMP BLINK_BEEP
                CHGVOL_TBLE:   ;:SELECT NEW VOL
178F 83         MOVC A,@A+PC
1790 22         RET
1791 20 14 40   DB 20H,20,40H,40,0,0,60H,60,0,0,10H,10
1794 28 00 00 60 3C 00 00 10 0A
                    :SHIFT TO NEW VOLUME

                CASE3:    ;:NO ACTION
179D 613F       AJMP BLINK_BEEP


                CASE2: ;NO ACTION
179F 613F       AJMP BLINK_BEEP

                CASE5:  ;:DISPLAY TEMP
17A1 10105B     JBC SEE_TEMP,RESTORE_VOL1
17A4 D210       SETB SEE_TEMP
17A6 C2D3       CLR PSW.3  ;BANKO
17A8 C2D4       CLR PSW.4
17AA ED         MOV A,RTEMP ;R5
17AB C3         CLR C
17AC 13         RRC A  ;DIV BY 2
17AD            BINARY_BCD DEC_HUN,DEC_TEN,DEC_ONE
+               ;CONVERTS BYTE LOCATED IN ACC TO  DECIMAL
+               ;AND STORES RESULT IN DEC_HUN, DEC_TEN AND ONE.
+17AD 752900     MOV DEC_HUN,#0  ;CLEAR REGISTERS
+17B0 752A00     MOV DEC_TEN,#0
+17B3 752B00     MOV DEC_ONE,#0
+               CALC_HUN0012:   ::SUBTRACT 100
+17B6 F5F0       MOV B,A
+               NEXTSUB10012:
+17B8 C3         CLR C
+17B9 9464       SUBB A,#100
+17BB 4006       JC CALC_TEN0012
+17BD 0529       INC DEC_HUN
+17BF F5F0       MOV B,A    ;SAVE
+17C1 80F5       SJMP NEXTSUB10012
+               CALC_TEN0012:   ;:SUBTRACT 10
+17C3 E5F0       MOV A,B
+               NEXTSUB20012:
+17C5 C3         CLR C
+17C6 940A       SUBB A,#10
+17C8 4006       JC CALC_ONE0012
+17CA 052A       INC DEC_TEN
+17CC F5F0       MOV B,A
+17CE 80F5       SJMP NEXTSUB20012
+               CALC_ONE0012:
+17D0 85F02B     MOV DEC_ONE,B
+17D3 E529       MOV A,DEC_HUN
+17D5 700A       JNZ BCD_OUT0012
+17D7 75290F     MOV DEC_HUN,#0FH   ;BLANK
+17DA E52A       MOV A,DEC_TEN
```

```
+17DC 7003      JNZ BCD_OUT0012
+17DE 752A0F    MOV DEC_TEN,#0FH   ;BLANK
+               BCD_OUT0012:
 17E1 00        NOP
 17E2 E529      MOV A,DEC_HUN
 17E4 C4        SWAP A
 17E5 F599      MOV SBUF,A
 17E7 D125      ACALL TRANS_DEL
 17E9 E52A      MOV A,DEC_TEN
 17EB C4        SWAP A
 17EC 4401      ORL A,#01H
 17EE F599      MOV SBUF,A
 17F0 D125      ACALL TRANS_DEL
 17F2 E528      MOV A,DEC_ONE
 17F4 C4        SWAP A
 17F5 4402      ORL A,#02H
 17F7 F599      MOV SBUF,A
 17F9 121625    LCALL TRANS_DEL
 17FC 02133F    LJMP BLINK_BEEP

                RESTORE_VOL1:  ;:DISPLAY VOL
 17FF C2D3      CLR PSW.3  ;BANK2
 1801 D2D4      SETB PSW.4
 1803 852899    MOV SBUF,CHG_VOL
 1806 121625    LCALL TRANS_DEL
 1809 759901    MOV SBUF,#01H
 180C 121625    LCALL TRANS_DEL
 180F 759902    MOV SBUF,#02H
 1812 121625    LCALL TRANS_DEL
                OUT_TEMP:
 1815 02133F    LJMP BLINK_BEEP

                CASE9:  ;:DISPLAY FLT LOAD
 1818 101113    JBC SEE_LD,RESTORE_VOL2
 181B D211      SETB SEE_LD
 181D 852C99    MOV SBUF,FLTLD_HUN
 1820 121625    LCALL TRANS_DEL
 1823 852D99    MOV SBUF,FLTLD_TEN
 1826 121625    LCALL TRANS_DEL
 1829 852E99    MOV SBUF,FLTLD_ONE
 182C 0140      AJMP OUT_DISPLD
                RESTORE_VOL2:  ;:DISPLAY VOL
 182E 852899    MOV SBUF,CHG_VOL
 1831 121625    LCALL TRANS_DEL
 1834 759901    MOV SBUF,#01H
 1837 121625    LCALL TRANS_DEL
 183A 759902    MOV SBUF,#02H
 183D 121625    LCALL TRANS_DEL
                OUT_DISPLD:
 1840 02133F    LJMP BLINK_BEEP

                CASEC:   ;:ENTER KEY
 1843 C204      CLR VEN_SEL
 1845 C203      CLR VOL_CHG
 1847 C23B      CLR TEMP
```

```
1849 C2D3       CLR PSW.3  :BANKO
184B C2D4       CLR PSW.4
184D 8E99       MOV SBUF,RVENT
184F 121625     LCALL TRANS_DEL
1852 852899     MOV SBUF,CHG_VOL  ;SET HUNS
1855 121625     LCALL TRANS_DEL
1858 759901     MOV SBUF,#01H  ;SET TENS
185B 121625     LCALL TRANS_DEL
185E 759902     MOV SBUF,#02H  ;SET ONES
1861 121625     LCALL TRANS_DEL
1864 02133F     LJMP BLINK_BEEP

                CASE7:   ;:NO ACTION
1867 02133F     LJMP BLINK_BEEP

                CASEB:   ;:NO ACTION
186A 02133F     LJMP BLINK_BEEP

                CASE6:   ;:SIL ALM 2 MIN
186D D2D3       SETB PSW.3  ;BANK1
186F C2D4       CLR PSW.4
1871 C2A7       CLR P2.7   ;OFF BUZZER
1873 D206       SETB SIL   ;SILENCE FLAG
1875 7E78       MOV RSIL_TIM,#120   ;R6,TWO MIN
1877 02133F     LJMP BLINK_BEEP

                CASEA:   ;:ALM RST
187A 75A078     MOV P2,#78H  ;OUTPUTS OFF
187D 752000     MOV 20H,#0  ;CLEAR BITS
1880 752100     MOV 21H,#0
1883 752200     MOV 22H,#0
1886 752700     MOV 27H,#0
1889 D200       SETB WAIT
188B D21E       SETB L16  ;WAIT
188D C21D       CLR L15  ;FILT CHANGE
188F C21F       CLR L17  ;LOFLOW
1891 852399     MOV SBUF,LED1
1894 31BE       ACALL TRANS_DEL1
1896 852899     MOV SBUF,CHG_VOL ;NORMAL LCD
1899 31BE       ACALL TRANS_DEL1
189B 759901     MOV SBUF,#01H
189E 31BE       ACALL TRANS_DEL1
18A0 759902     MOV SBUF,#02H
18A3 31BE       ACALL TRANS_DEL1
18A5 C2D3       CLR PSW.3 ;BANKO
18A7 C2D4       CLR PSW.4
18A9 8E99       MOV SBUF,RVENT ;R1
18AB 31BE       ACALL TRANS_DEL1
18AD 53260F     ANL LED2,#0FH ;OFF
18B0 852699     MOV SBUF,LED2
18B3 31BE       ACALL TRANS_DEL1
18B5 C22C       CLR L34  :HI PRESS
18B7 852599     MOV SBUF,LED3
18BA 31BE       ACALL TRANS_DEL1
18BC 752F00     MOV THREE_CYCLE,#0  :RESET
```

```
18BF D2D3       SETB PSW.3  :BANK1
18C1 C2D4       CLR PSW.4
18C3 75??00     MOV RON_TIM,#0
18C5 75??00     MOV ROFF_TIM,#0
18C7 75????     MOV POSSUM.#0
18CA 75????     MOV NEGSUM,#0
18CD 75????     MOV TH0,#70H  :RST TIMER
18D0 75A8??     MOV IE,#87H   ;SET ET0
18D3 758???     MOV TCON,#50H ;SET TR0
18D6 22         RET

                CASED:   ;:DISPLAY PIP
                  ;MULTIPLY BY SCALE FACTOR OF 5/8, CONVERT TO BCD
                  ;AND DISPLAY PIP. RETURN TO VOLUME DISPLAY WHEN
                  ;SWITCH IS PRESSED A SECOND TIME.

18D7 10??6B     JBC SEE_PIP,LCD_VOL
18DA D2??       SETB SEE_PIP
18DC E54E       MOV A,PIP_STORE
18DE 75F?05     MOV B,#5
18E1 A4         MUL AB    :MSB IN B
18E2 C5F0       XCH A,B   ;RRC 3 TIMES TO DIVIDE BY 8
18E4 13         RRC A      ;MSB IN A
18E5 C5F0       XCH A,B   :LSB IN A
18E7 13         RRC A
18E8 C3         CLR C     ;SECOND ROTATION
18E9 C5F0       XCH A,B
18EB 13         RRC A
18EC C5F0       XCH A,B
18EE 13         RRC A
18EF C3         CLR C     ;THIRD ROTATION
18F0 C5F0       XCH A,B
18F2 13         RRC A
18F3 C5F0       XCH A,B
18F5 13         RRC A
18F6 94??       SUBB A.#14H  ;ZERO OFFSET
18F8            BINARY_BCD DEC_HUN,DEC_TEN,DEC_ONE
+               :CONVERTS BYTE LOCATED IN ACC TO  DECIMAL
+               ;AND STORES RESULT IN DEC_HUN, DEC_TEN AND ONE.
+18F8 75????     MOV DEC_HUN,#0  ;CLEAR REGISTERS
+18FB 75????     MOV DEC_TEN,#0
+18FE 75????     MOV DEC_ONE,#0
+               CALC_HUN0013:  ::SUBTRACT 100
+1901 F5F0       MOV B,A
+               NEXTSUB10013:
+1903 C3         CLR C
+1904 94??       SUBB A,#100
+1906 40??       JC CALC_TEN0013
+1908 05??       INC DEC_HUN
+190A F5F0       MOV B,A     :SAVE
+190C 80F?       SJMP NEXTSUB10013
+               CALC_TEN0013:  ::SUBTRACT 10
+190E E5F0       MOV A,B
+               NEXTSUB20013:
+1910 C3         CLR C
```

```
+1911 940A      SUBB A,#10
+1913 4006      JC CALC_ONE0013
+1915 052A      INC DEC_TEN
+1917 F5F0      MOV B,A
+1919 80F5      SJMP NEXTSUB20013
+               CALC_ONE0013:
+191B 85F02B    MOV DEC_ONE,B
+191E E529      MOV A,DEC_HUN
+1920 700A      JNZ BCD_OUT0013
+1922 75290F    MOV DEC_HUN,#0FH   ;BLANK
+1925 E52A      MOV A,DEC_TEN
+1927 7003      JNZ BCD_OUT0013
+1929 752A0F    MOV DEC_TEN,#0FH   ;BLANK
+               BCD_OUT0013:
 192C E529      MOV A,DEC_HUN ;DISPLAY PIP
 192E C4        SWAP A
 192F F599      MOV SBUF,A
 1931 31BE      ACALL TRANS_DEL1
 1933 E52A      MOV A,DEC_TEN
 1935 C4        SWAP A
 1936 4401      ORL A,#01H
 1938 F599      MOV SBUF,A
 193A 31BE      ACALL TRANS_DEL1
 193C E52B      MOV A,DEC_ONE
 193E C4        SWAP A
 193F 4402      ORL A,#02H
 1941 F599      MOV SBUF,A
 1943 800D      SJMP OUTPIP
                LCD_VOL:   ;:DISPLAY VOL
 1945 852899    MOV SBUF,CHG_VOL
 1948 31BE      ACALL TRANS_DEL1
 194A 759901    MOV SBUF,#01H
 194D 31BE      ACALL TRANS_DEL1
 194F 759902    MOV SBUF,#02H
                OUTPIP:
 1952 02133F    LJMP BLINK_BEEP

                CASEE:    ;:ALM TEST
                ;PUSH SW TO TEST & PUSH TO RETURN
 1955 100A33    JBC ALM_TST,NORMAL
 1958 D20A      SETB ALM_TST
 195A D2A7      SETB P2.7    ;ON BUZZER
 195C 7480      MOV A,#80H
 195E F599      MOV SBUF,A   ;HUNS LCD TEST
 1960 1219BE    LCALL TRANS_DEL1
 1963 04        INC A        ;TENS
 1964 F599      MOV SBUF,A
 1966 1219BE    LCALL TRANS_DEL1
 1969 04        INC A        ;ONES
 196A F599      MOV SBUF,A
 196C 1219BE    LCALL TRANS_DEL1
 196F 04        INC A        ;VENT #
 1970 F599      MOV SBUF,A
 1972 1219BE    LCALL TRANS_DEL1
 1975 74F4      MOV A,#0F4H ;LED1 TEST
```

```
1977 F599      MOV SBUF,A
1979 1219BE    LCALL TRANS_DEL1
197C 04        INC A        ;LED2
197D F599      MOV SBUF,A
197F 1219BE    LCALL TRANS_DEL1
1982 04        INC A
1983 F599      MOV SBUF,A    ;LED3
1985 1219BE    LCALL TRANS_DEL1
1988 02198A    LJMP OUT_TST
               NORMAL:   ;:NORMAL DISPLAY
198B C2A7      CLR P2.7    ;RESTORE ALARM & DISPLAYS
198D C2D3      CLR PSW.3   :BANK0
198F C2D4      CLR PSW.4
1991 852899    MOV SBUF,CHG_VOL
1994 1219BE    LCALL TRANS_DEL1
1997 759901    MOV SBUF,#01H
199A 1219BE    LCALL TRANS_DEL1
199D 759902    MOV SBUF,#02H
19A0 1219BE    LCALL TRANS_DEL1
19A3 8E99      MOV SBUF,RVENT
19A5 1219BE    LCALL TRANS_DEL1
19A8 852399    MOV SBUF,LED1
19AB 1219BE    LCALL TRANS_DEL1
19AE 852699    MOV SBUF,LED2
19B1 1219BE    LCALL TRANS_DEL1
19B4 852599    MOV SBUF,LED3
19B7 1219BE    LCALL TRANS_DEL1
               OUT_TST:
198A 02133F    LJMP BLINK_BEEP
198D 00        NOP

               TRANS_DEL1:   ;:DELAY 2.25MS,CC=80EH
198E D51DFD    DJNZ DIVIDE1,TRANS_DEL1   ;COUNT 255
19C1 751DFF    MOV DIVIDE1,#0FFH   :RESET
19C4 D51EF7    DJNZ DIVIDE2,TRANS_DEL1   ;COUNT 4
19C7 751E04    MOV DIVIDE2,#04H   ;RESET
19CA 22        RET
19CB 00        NOP

               MAN_SW:   ;:ON VALVES
19CC C0E0      PUSH ACC
19CE C0D0      PUSH PSW
19D0 C2A8      CLR EX0   :DISABLE INT
19D2 53A08F    ANL P2,#10001111B ;ON VALVES
19D5 12156E    HOLDIT: LCALL SERVICE
19D8 30B2FA    JNB P3.2,HOLDIT
19DB D200      SETB WAIT
19DD D21E      SETB L16   ;WAIT LED
19DF 852399    MOV SBUF,LED1
19E2 31BE      ACALL TRANS_DEL1
19E4 43A070    ORL P2,#01110000B   ;OFF VALVES
19E7 752F00    MOV THREE_CYCLE,#00H
19EA D2A8      SETB EX0   :ENABLE INT0
19EC D0D0      POP PSW
19EE D0E0      POP ACC
```

```
19F0 32          RETI

                 ;%E
19F1             ENDS    :CODE SEGMENT
1000             END   BEGIN
```

| ;%T Symbol Name | Type | Value |
|---|---|---|
| ALARM | L | 112F |
| ALARM1 | L | 11AB |
| ALARM2 | L | 12A4 |
| ALM | B | 000C |
| ALM_TST | B | 000A |
| ANALOG | M | 0000 |
| AWP_AVG | D | 004C |
| AWP_LO | D | 0048 |
| AWP_MAX | D | 004D |
| BANK0 | U | 0000 |
| BCD_OUT0011 | L | 16DD |
| BCD_OUT0012 | L | 17E1 |
| BCD_OUT0013 | L | 192C |
| BEEP | B | 0014 |
| BEGIN | L | 1000 |
| BINARY_BCD | M | 0000 |
| BLINK_BEEP | L | 133F |
| BLINK_BEEP1 | L | 1623 |
| CALC_HUN0011 | L | 16B2 |
| CALC_HUN0012 | L | 17B6 |
| CALC_HUN0013 | L | 1901 |
| CALC_ONE0011 | L | 16CC |
| CALC_ONE0012 | L | 17D0 |
| CALC_ONE0013 | L | 191B |
| CALC_PIP | L | 115D |
| CALC_TEMP | L | 16A3 |
| CALC_TEN0011 | L | 16BF |
| CALC_TEN0012 | L | 17C3 |
| CALC_TEN0013 | L | 190E |
| CASE0 | L | 167D |
| CASE1 | L | 16F4 |
| CASE2 | L | 179F |
| CASE3 | L | 179D |
| CASE4 | L | 1726 |
| CASE5 | L | 17A1 |
| CASE6 | L | 186D |
| CASE61 | L | 165D |
| CASE7 | L | 1867 |
| CASE71 | L | 1661 |
| CASE8 | L | 1769 |
| CASE81 | L | 1765 |
| CASE9 | L | 1818 |
| CASE91 | L | 1665 |
| CASEA | L | 187A |
| CASEA1 | L | 1669 |
| CASEB | L | 186A |
| CASEB1 | L | 166D |
| CASEC | L | 1843 |
| CASEC1 | L | 1671 |
| CASED | L | 18D7 |
| CASED1 | L | 1675 |
| CASEE | L | 1955 |

```
CASEE1. . . . . L 1679
CASEF . . . . . L 165A
CHARGE. . . . . L 11B4
CHGVOL_TBLE . . . . . L 178F
CHG_VOL . . . . . D 0028
CHK_AWP . . . . . L 1107
CHK_CHGTIM. . . . . L 11E6
CHK_DPTHRESH. . . . . L 1242
CHK_EOEXH . . . . . L 12A7
CHK_EOEXH1. . . . . L 123F
CHK_EXH . . . . . L 1100
CHK_HOLD. . . . . L 1429
CHK_LED21 . . . . . L 1358
CHK_LED22 . . . . . L 13A9
CHK_LED31 . . . . . L 1363
CHK_LED32 . . . . . L 13B4
CHK_OFFTIM. . . . . L 15ED
CHK_PEAK. . . . . L 1150
CHK_SERPORT . . . . . L 162B
CHK_SIL . . . . . L 1134
CHK_VOL . . . . . L 11C6
CHK_VOL1. . . . . L 136E
CHK_VOL2. . . . . L 13BF
CHK_WAIT. . . . . L 1308
CHK_WAIT1 . . . . . L 11E3
CLEAR . . . . . L 1407
CLK . . . . . B 0039
CLOG1 . . . . . B 0016
CLOG2 . . . . . B 0017
CLOG_HI . . . . . D 0046
CLOG_LO . . . . . D 0045
CONT. . . . . L 113C
CONT1 . . . . . L 1555
CONT2 . . . . . L 15CA
CONT3 . . . . . L 15D4
CONT4 . . . . . L 15DF
CONT5 . . . . . L 10D2
CONT6 . . . . . L 141A
DEC_HUN . . . . . D 0029
DEC_ONE . . . . . D 002B
DEC_TEN . . . . . D 002A
DEL1. . . . . B 0012
DELAY1. . . . . L 10C6
DELAY2. . . . . L 10CB
DELAY5. . . . . L 1122
DEL_4TENTHS . . . . . B 0013
DISPLAY_TEMP. . . . . L 16A7
DIV21 . . . . . B 0002
DIV22 . . . . . B 000B
DIV23 . . . . . B 0038
DIV24 . . . . . B 000E
DIVIDE1 . . . . . D 001D
DIVIDE2 . . . . . D 001E
END_DEL . . . . . L 10D0
EOI . . . . . L 10EB
```

```
EXH . . . . . B 0001
FIFO. . . . . M 0EB4
FILTAWP_THRESH. . . . . I 0037
FILTDP_THRESH . . . . . I 008D
FIRST_SAMP. . . . . L 1220
FLO . . . . . L 15DB
FLOTIM. . . . . I 000B
FLOW. . . . . B 000F
FLO_ALM . . . . . L 160E
FLO_TH. . . . . I 002D
FLO_TST . . . . . L 15B5
FLTFLO_AVG. . . . . D 0044
FLTFLO_LO . . . . . D 0040
FLTLD25 . . . . . D 0055
FLTLD50 . . . . . D 0056
FLTLD75 . . . . . D 0057
FLTLD_HUN . . . . . D 002C
FLTLD_ONE . . . . . D 002E
FLTLD_TEN . . . . . D 002D
FLT_LD. . . . . L 1225
GO_ON . . . . . L 1339
HEAT. . . . . B 003A
HEAT_CHK. . . . . L 1592
HIPRESS . . . . . L 11A3
HITEMP. . . . . L 157B
HI_TEMP . . . . . L 15AB
HOLD. . . . . B 0008
HOLDIT. . . . . L 19D5
HOLD_OFF. . . . . L 170B
HOLD_OUT. . . . . L 1716
INC3. . . . . L 1336
INIT1 . . . . . L 1032
INITIALIZE. . . . . L 1030
INSP. . . . . B 0015
JUMP_TBLE1. . . . . L 1639
L14 . . . . . B 001C
L15 . . . . . B 001D
L16 . . . . . B 001E
L17 . . . . . B 001F
L24 . . . . . B 0034
L25 . . . . . B 0035
L26 . . . . . B 0036
L27 . . . . . B 0037
L34 . . . . . B 002C
L35 . . . . . B 0020
LCD_VOL . . . . . L 1945
LED1. . . . . D 0023
LED2. . . . . D 0026
LED3. . . . . D 0025
MAIN_LOOP . . . . . L 10D8
MAN_SW. . . . . L 19CC
NEGFLO. . . . . L 154A
NEGSUM. . . . . D 0051
NEW_TEMP. . . . . L 1699
NEXT10001 . . . . . L 116D
```

```
NEXT10002 . . . . . L 12C9
NEXT10004 . . . . . L 145D
NEXT10006 . . . . . L 14A6
NEXT10008 . . . . . L 14EF
NEXT20001 . . . . . L 1185
NEXT20002 . . . . . L 12E1
NEXT20004 . . . . . L 1474
NEXT20006 . . . . . L 148D
NEXT20008 . . . . . L 1506
NEXTSUB10011. . . . . L 16B4
NEXTSUB10012. . . . . L 17B8
NEXTSUB10013. . . . . L 1903
NEXTSUB20011. . . . . L 16C1
NEXTSUB20012. . . . . L 17C5
NEXTSUB20013. . . . . L 1910
NEXT_SAMP . . . . . L 1524
NOFLO . . . . . L 15D0
NOFLOTIM. . . . . I 0032
NOFLO_ALM . . . . . L 15F7
NOFLO_TH. . . . . I 008C
NORMAL. . . . . L 198B
NOT_EQ. . . . . L 1587
OFFTIMER. . . . . D 0015
OFF_ALM . . . . . B 000D
OFF_STATE . . . . . L 1687
ONTIMER . . . . . D 0014
OUT . . . . . L 13EB
OUT1. . . . . L 133C
OUTPIP. . . . . L 1952
OUT_DISPLD. . . . . L 1840
OUT_TEMP. . . . . L 1815
OUT_TST . . . . . L 198A
PATINSP_THRESH. . . . . I 0008
PAT_INSP. . . . . L 12F6
PEEP_AVG. . . . . D 0063
PEEP_LO . . . . . D 0060
PIP_AVG . . . . . D 005B
PIP_LO. . . . . D 0058
PIP_STORE . . . . . D 004E
PIP_THRESH. . . . . I 00E0
POSSUM. . . . . D 0050
RAW_PRESS . . . . . R 0003
RCHG_TIM. . . . . R 0001
RDIV10. . . . . R 0002
RDIV5 . . . . . R 0003
RESTORE_VOL1. . . . . L 17FF
RESTORE_VOL2. . . . . L 182E
RETURN. . . . . L 1567
RFLT_FLO. . . . . R 0002
RHOLD_TIM . . . . . R 0007
RNEB_FLO. . . . . R 0004
ROFF_TIM. . . . . R 0005
RON_TIM . . . . . R 0004
RSIL_TIM. . . . . R 0006
RTEMP . . . . . R 0005
```

```
RUNNING_AVG . . . . . M 0000
RVENT . . . . . R 0006
RVENT_SIG . . . . . R 0001
SAMPLE. . . . . L 1435
SEE_LD. . . . . B 0011
SEE_PIP . . . . . B 0009
SEE_TEMP. . . . . B 0010
SEE_VENT. . . . . L 1743
SERVICE . . . . . L 156E
SET_CHGTIM. . . . . D 0019
SET_CLK . . . . . L 1413
SET_EXH . . . . . L 1117
SIL . . . . . B 0006
SPON_BR . . . . . B 0007
STRT_EXH. . . . . L 11AE
SW_OFF. . . . . L 15A1
TEMP. . . . . B 003B
TEMP_CONT . . . . . L 1573
TEMP_DEC. . . . . D 0012
TEMP_HI . . . . . I 00A0
TEMP_SET. . . . . D 0011
TEMP_STORE. . . . . D 001C
TEST0 . . . . . L 1298
TEST25. . . . . L 1286
TEST50. . . . . L 1274
TEST75. . . . . L 1258
THREE_CYCLE . . . . . D 002F
TIM_SAMP. . . . . L 13F3
TRANS_DEL . . . . . L 1625
TRANS_DEL1. . . . . L 198E
TST_BEEP1 . . . . . L 1394
TST_BEEP2 . . . . . L 13E6
TST_TEMP1 . . . . . L 1387
TST_TEMP2 . . . . . L 13D9
TST_VENTSEL1. . . . . L 1380
TST_VENTSEL2. . . . . L 13D1
TURN_OFF. . . . . L 139E
TURN_ON . . . . . L 134E
VENT_AVG. . . . . D 006C
VENT_HI . . . . . D 001B
VENT_LO . . . . . D 001A
VENT_LOW. . . . . D 0068
VENT_TBLE . . . . . L 175D
VEN_SEL . . . . . B 0004
VOL_CHG . . . . . B 0003
WAIT. . . . . B 0000

:%Z

00 Errors  (0000)
```

WHAT IS CLAIMED IS:

1. A nebulizer comprising:

a housing containing a reservoir for holding a liquid to be nebulized and an air space above the reservoir for holding aerosol;

means for generating said aerosol by nebulizing said liquid;

means for attaching said housing to a mechanical respirator having an inhalation phase, an exhalation phase, a gas flow passageway to a patient, and an external electrical signal source capable of generating a first electrical signal during said exhalation phase;

means responsive to said first electrical signal for introducing said aerosol into said gas flow passageway, such that said aerosol fills said gas flow passageway during a portion of said exhalation phase.

2. The nebulizer of Claim 1 further comprising means for monitoring the amount of said aerosol introduced into said gas flow passageway.

3. The nebulizer of Claim 1 wherein said mechanical respirator further being capable of generating a second electrical signal during said inhalation phase.

4. The nebulizer of Claim 3 wherein said aerosol generating means further comprising a plurality of nebulizer nozzles each having means for controlling the gas flow therethrough.

5. The nebulizer of Claim 4, wherein said introducing means further comprises:

a gas flow for directing compressed gas from a compressed gas source to each of said plurality of controlling means for said nebulizer nozzles; said gas flow means including means responsive to said first electrical signal for opening a conduit of said nebulizer nozzles and for closing the conduit to said nebulizer nozzles simultaneously or one at a time, in response to said second electrical signal.

6. The nebulizer of Claim 5 further comprising:

means responsive to said second electrical signal for generating a decreasing flow of gas; and

means for directing said decreasing flow of gas into said mechanical respirator.

7. A method of operating a nebulizer of the type having means for generating an aerosol and means for supplying said aerosol to a mechanical respirator having an inhalation phase, an exhalation phase and a gas passageway to a patient, and an external electrical signal source capable of generating a first electrical signal during said exhalation phase, method comprising:

generating said aerosol; and

introducing said aerosol into said gas passageway during a portion or all of the said exhalation phase.

8. The method of Claim 7 wherein said introducing step further comprising:

opening a valve, in response to said first signal, to introduce said aerosol from said nebulizer to said gas passageway.

9. The method of Claim 7 wherein said generating step further comprises:

entraining a liquid into a source of compressed gas to generate said aerosol, in response to said first signal and continuing until standardized volume of aerosol dose has been delivered.

10. The method of Claim 7 wherein said external electrical signal source is capable of generating a second electrical signal during said inhalation phase.

11. The method of Claim 10 further comprising:

ceasing the generation of said aerosol in response to said second electrical signal.

12. A nebulizer for use with a respirator means having an inhalation phase and an exhalation phase, a first tubing means connecting said respirator means with a patient wherein during said inhalation phase said respirator means is fluidically connected to said patient through said first tubing means for introducing breathing gas in said first tubing means into respiratory tract of the said patient, a second tubing means connecting said respirator means with said patient wherein during said exhalation phase said respirator means is fluidically connected to said patient through said second tubing means for receiving exhaled gas from said patient to said respirator means, said respirator means further

having means for generating a first electrical signal during said exhalation phase; said nebulizer comprising:

means for generating an aerosol;

aerosol connecting means for connecting said generating means to said first tubing means; and

means for introducing said aerosol into said first tubing means in response to and synchronized with said first electrical signal.

13. The nebulizer of Claim 12 further comprising:

housing means containing a reservoir for holding a liquid to be nebulized and an air space above the reservoir for holding said aerosol.

14. The nebulizer of Claim 13 wherein said aerosol connecting means connects said air space to said first tubing means.

15. The nebulizer of Claim 14 wherein said generating means comprising:

a plurality of nebulizing nozzles each having means for controlling the gas flow therethrough.

16. The nebulizer of Claim 15 wherein said respirator means for generating a second electrical signal during said inhalation phase.

17. The nebulizer of Claim 16 wherein said introducing means for all of said nebulizing nozzles, in response to said first electrical signal, de-

activates said controlling means, either simultaneously or one at a time.

18. The nebulizer of Claim 14 further comprising means for monitoring said aerosol introduced into said first tubing means.

19. The nebulizer of Claim 16 further comprising:

means for generating a decreasing flow of gas; and

means for directing said decreasing volume of gas into said second tubing means.

20. The nebulizer of Claim 12 wherein said means for generating said first electrical signal further comprises:

a filter pressure sensor for detecting the pressure differential in said second tubing means, and for generating a filter pressure signal in response thereto;

an airway pressure sensor for detecting the pressure in said first tubing means, and for generating an airway pressure signal in response thereto; and

means for receiving said filter pressure signal and said airway pressure signal and for generating said first electrical signal synchronized with the commencement of said exhalation phase.

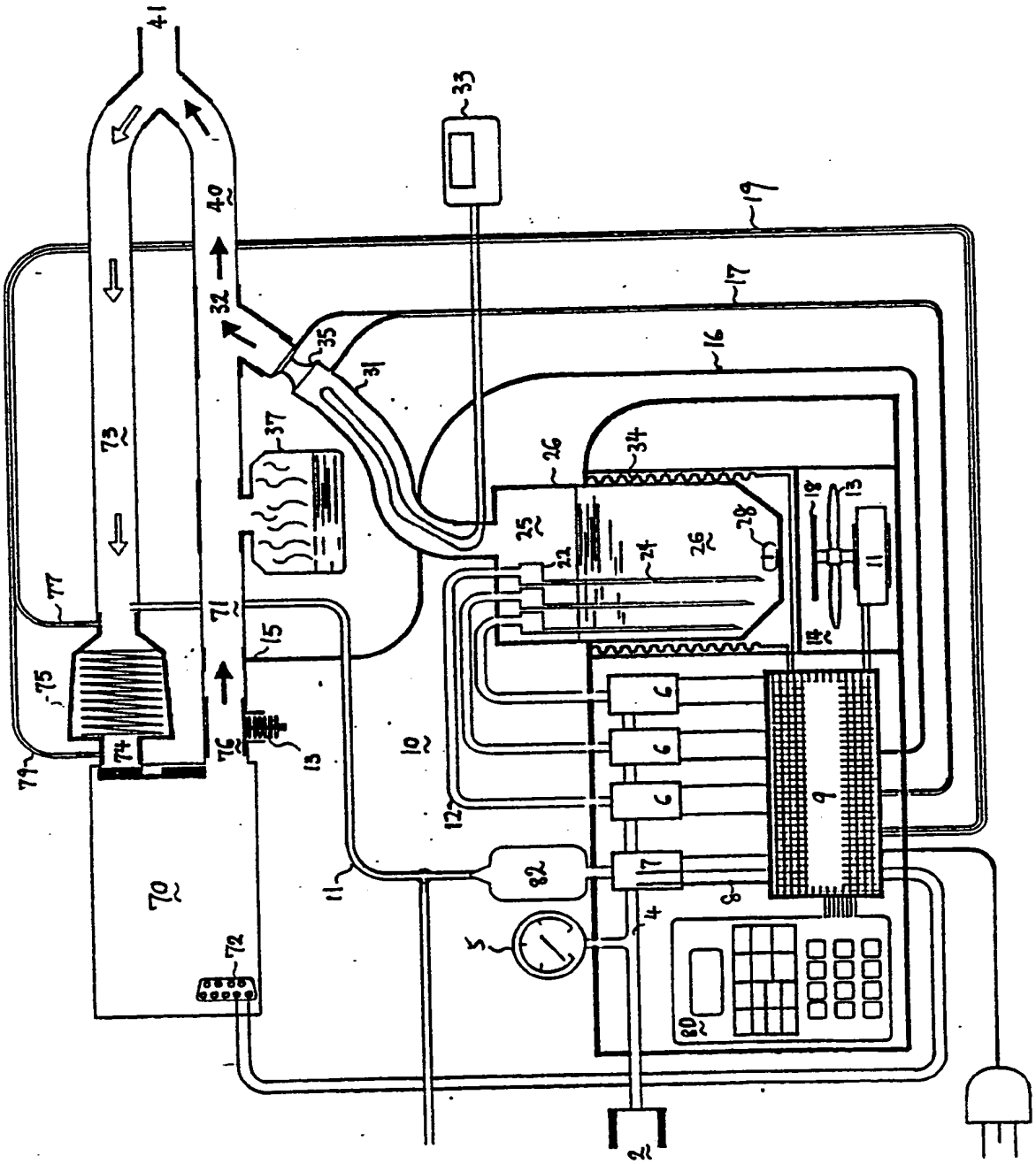



FIGURE 1

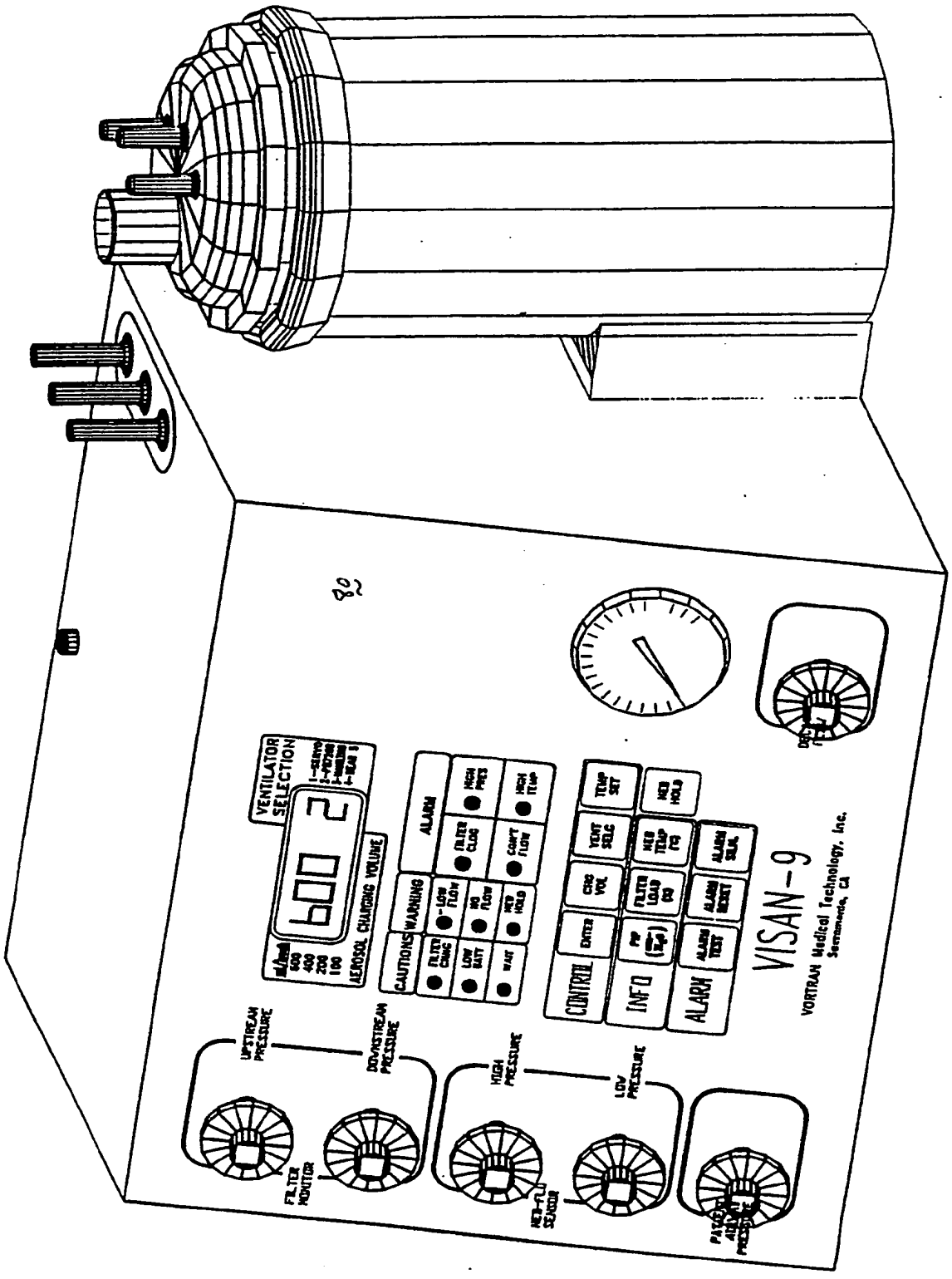

VENTILATOR SELECTION
AEROSOL CHARGING VOLUME
ALARM
WARNING
CAUTIONS
CONTROL
INFO
ALARM
VISAN-9
VORTRAN Medical Technology, Inc.
Sacramento, CA
UPSTREAM PRESSURE
DOWNSTREAM PRESSURE
HIGH PRESSURE
LOW PRESSURE
FILTER MONITOR

FIGURE 2

# INTERNATIONAL SEARCH REPORT

International Application No. PCT/US92/00566

## I. CLASSIFICATION OF SUBJECT MATTER (if several classification symbols apply, indicate all) [6]

According to International Patent Classification (IPC) or to both National Classification and IPC

I.P.C. (5): A61M 15/00, A61M 16/10, A62B 7/00, F16K 31/02
U.S. Cl. : 128/203.12, 204.21, 204.23, 204.26

## II. FIELDS SEARCHED

Minimum Documentation Searched [7]

| Classification System | Classification Symbols |
|---|---|
| U.S. | 128/200 14, 200.21; 203.12, 203.13, 203.14, 203.16, 203.17 203.26, 203.27, 204.17, 204.18, 204.21, 204.23, 204.26 |

Documentation Searched other than Minimum Documentation to the Extent that such Documents are Included in the Fields Searched [8]

## III. DOCUMENTS CONSIDERED TO BE RELEVANT [9]

| Category * | Citation of Document, [11] with indication, where appropriate, of the relevant passages [12] | Relevant to Claim No. [13] |
|---|---|---|
| Y | US, A, 4,106,503 (ROSENTHAL et al) 15 AUGUST 1978 See entire document | 1-3,7-14,18-20 |
| Y | US, A, 4,832,014 (PERKINS) 23 MAY 1989 See entire document | 1-3,7-14,18-20 |
| Y | US, A, 4,197,843 (BIRD) 15 APRIL 1980 See entire document | 1-3,7-14,18-20 |
| Y | US, A, RESPIRATORY THERAPY EQUIPMENT (MCPHERSON) @1985, C.V. MOSBY CO., pp. 128-131, 158-163, 468-469, 476-479 & 442-443 | 1-3,7-14,18-20 |

* Special categories of cited documents: [10]

"A" document defining the general state of the art which is not considered to be of particular relevance

"E" earlier document but published on or after the international filing date

"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step

"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art.

"&" document member of the same patent family

## IV. CERTIFICATION

| Date of the Actual Completion of the International Search | Date of Mailing of this International Search Report |
|---|---|
| 27 APRIL 1992 | 27 MAY 1992 |
| **International Searching Authority** | **Signature of Authorized Officer** |
| ISA/US | KIMBERLY L. ASHER |

Form PCT/ISA/210 (second sheet) (Rev. 11-87)

**CORRECTED VERSION***

**PCT**

WORLD INTELLECTUAL PROPERTY ORGANIZATION
International Bureau

INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

| (51) International Patent Classification 5 : | | (11) International Publication Number: | WO 92/12750 |
|---|---|---|---|
| A61M 15/00, 16/10, A62B 7/00<br>F16K 31/02 | A1 | (43) International Publication Date: | 6 August 1992 (06.08.92) |

(21) International Application Number: PCT/US92/00566

(22) International Filing Date: 22 January 1992 (22.01.92)

(30) Priority data:
645,579 24 January 1991 (24.01.91) US

(71) Applicant: VORTRAN MEDICAL TECHNOLOGY, INC. [US/US]; 3941 J Street, Suite 354, Sacramento, CA 95819-3633 (US).

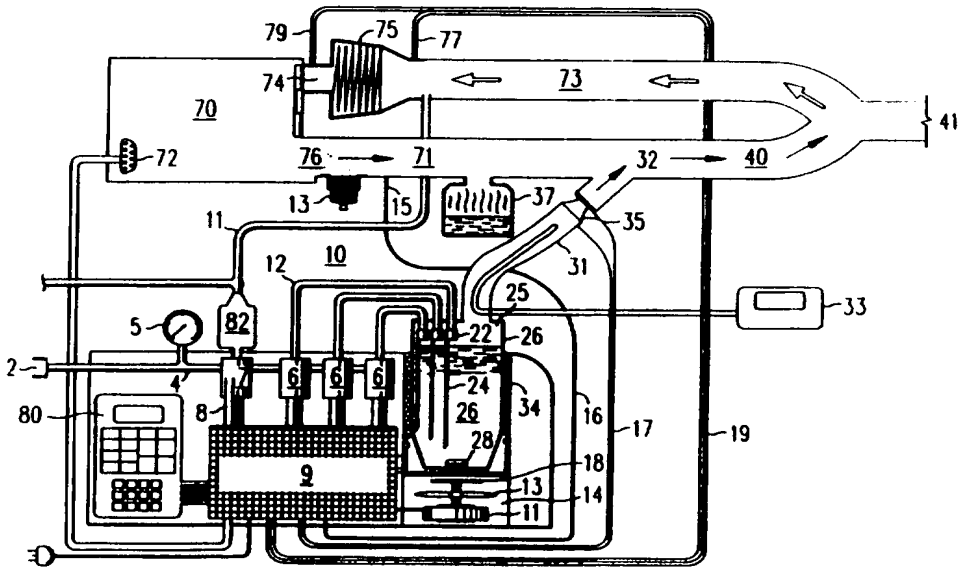
(72) Inventors: RAABE, Otto, G. ; 652 Buchanan Street, Davis, CA 95616 (US). LEE, James, I., C. ; 7861 Rush River Drive, Sacramento, CA 95831 (US). HATHAWAY, James, Calvin ; 441 Hartnell Place, Sacramento, CA 95825 (US).

(74) Agents: YIN, Ronald, L. et al.; Limbach & Limbach, 2001 Ferry Building, San Francisco, CA 94111 (US).

(81) Designated States: AT (European patent), AU, BE (European patent), CA, CH (European patent), DE (European patent), DK (European patent), ES (European patent), FR (European patent), GB (European patent), GR (European patent), IT (European patent), JP, LU (European patent), MC (European patent), NL (European patent), NO, SE (European patent).

**Published**
*With international search report.*
*Before the expiration of the time limit for amending the claims and to be republished in the event of the receipt of amendments.*

(54) Title: INTERMITTENT SIGNAL ACTUATED NEBULIZER SYNCHRONIZED WITH EXHALATION




(57) Abstract

A self-contained, high capacity nebulizer (10), having automatic mixing (28) and temperature control (34) features is provided. The nebulizer is designed for use in conjunction with mechanical respirators (70), ventilators, or breathing machines, and for this purpose will use electrical signals (8) generated by or received from the respirator (70) to automatically control and synchronize the nebulizing and mixing functions such that nebulization occurs only during the exhalation phase of the respiratory function to load the gas passageway of the respirator (70) to the patient with a standardized dose of medicinal aerosol. Upon commencement of the inhalation phase, the aerosol in the gas passageway is ventilated into the lungs of the patient to which it is attached.

* (Referred to in PCT Gazette No. 24/1992, Section II)

***FOR THE PURPOSES OF INFORMATION ONLY***

Codes used to identify States party to the PCT on the front pages of pamphlets publishing international applications under the PCT.

| | | | | | |
|---|---|---|---|---|---|
| **AT** | Austria | **FI** | Finland | **ML** | Mali |
| **AU** | Australia | **FR** | France | **MN** | Mongolia |
| **BB** | Barbados | **GA** | Gabon | **MR** | Mauritania |
| **BE** | Belgium | **GB** | United Kingdom | **MW** | Malawi |
| **BF** | Burkina Faso | **GN** | Guinea | **NL** | Netherlands |
| **BG** | Bulgaria | **GR** | Greece | **NO** | Norway |
| **BJ** | Benin | **HU** | Hungary | **PL** | Poland |
| **BR** | Brazil | **IE** | Ireland | **RO** | Romania |
| **CA** | Canada | **IT** | Italy | **RU** | Russian Federation |
| **CF** | Central African Republic | **JP** | Japan | **SD** | Sudan |
| **CG** | Congo | **KP** | Democratic People's Republic of Korea | **SE** | Sweden |
| **CH** | Switzerland | | | **SN** | Senegal |
| **CI** | Côte d'Ivoire | **KR** | Republic of Korea | **SU** | Soviet Union |
| **CM** | Cameroon | **LI** | Liechtenstein | **TD** | Chad |
| **CS** | Czechoslovakia | **LK** | Sri Lanka | **TG** | Togo |
| **DE** | Germany | **LU** | Luxembourg | **US** | United States of America |
| **DK** | Denmark | **MC** | Monaco | | |
| **ES** | Spain | **MG** | Madagascar | | |

# INTERMITTENT SIGNAL ACTUATED NEBULIZER SYNCHRONIZED WITH EXHALATION

This application is a continuation-in-part of copending U.S. Patent Application Serial No. 07/585,616, filed on September 20, 1990, which is a continuation of U.S. Patent Application Serial No. 270,520, filed on November 14, 1988, now abandoned, which is a continuation of U.S. Patent Application Serial No. 07/071,202, filed on July 8, 1987, now U.S. Patent 4,832,012.

## Technical Field

The present invention relates to nebulizers for creating medicinal aerosols for inhalation therapy. In particular, the present invention relates to nebulizers used during the exhalation phase of the breathing cycle in conjunction with and without interfering with mechanical breathing machines which are used to ventilate the lungs of patients who cannot breathe unaided.

## Background Art

The thin membrane of the lungs provides an easily penetrated, convenient and generally safe means for obtaining rapid absorption of medication by the body. This is especially desirable where the lungs themselves are diseased or injured. Such medication or drugs are generally delivered to the lung membrane in the form of a fine mist or aerosol which is breathed into the lungs through the nose or mouth of the patient. A variety of devices, called nebulizers by those skilled in the art, have been developed for converting liquids into fine aerosols

for this purpose. The simplest of these devices is the hand-held atomizer which converts a liquid to an aerosol when a bulb is compressed to produce a jet of air which atomizes the medication and propels it out of the atomizer. To be effective, the aerosols need to be provided at high concentrations and with droplet size in the respirable range (mass median aerodynamic diameter less than 3 micrometers).

Nebulizers are particularly useful for initiating and continuing respiratory therapy in conjunction with respirators, mechanical ventilators or breathing machines (hereinafter referred to generically as respirators) used to ventilate the lungs of patients having serious respiratory impairment. While some respirators incorporate nebulizers in their design, many do not. Nebulizers incorporated into the structure of such respirators often suffer from many disadvantages. One such disadvantage is severely limited capacity for medication to be nebulized, requiring frequent interruptions in the therapy as new medication is added to the nebulizer reservoir.

Most, if not all, such nebulizers are incorporated in respirators in which the inhalation and exhalation phases of the breathing cycle are triggered by changes in air pressure caused by the patient himself. Such "demand" respirators are not useful for patients whose respiratory systems are paralyzed and incapable of causing even slight changes in air pressure. These patients are aided by mechanical respirators in which the phases of the breathing cycle are triggered by electrical signals. There is now no effective means for patients on such respirators to receive aerosol treatment.

Thus, the need exists for a nebulizer which can be attached to a mechanical respirator, especially those in which the breathing cycle is controlled by an electrical signal, which has a reservoir capacity sufficient to enable several hours of continuous treatment, which can prevent the settling of suspensions or mixtures without creating nebulization-destroying turbulence.

U.S. Patent 4,832,012 discloses the principal of signal actuated synchronization of nebulization for delivery of aerosolized medicine to patients whose breathing is supported or augmented by a mechanical respiratory. In that reference, nebulization could be effected during inhalation or exhalation, but the primary trust of that reference was to provide aerosols during the inhalation phase of the breathing cycle to mix with the inhalation tidal volume provided by the respirator, and in synchrony with the normal operation of the respiratory. However, it has been found that the addition of volume of gas to mix with the inhalation tidal volume provided by the respirator, may interfere with the normal operation of the respirator in certain operating modes, and the medicinal aerosol is diluted by the portion of gas delivered by the respirator.

## Summary of the Invention

The present invention is based upon the nebulization of medicine during and synchronized with the exhalation portion of each breath of the breathing cycle to fill the airline leading from the nebulizer to the patient with a standardized dose of medicinal aerosols that are delivered to the lung by the force of the flow of breathing gas (oxygen-enriched air) delivered by the respirator during the

inhalation portion of the breathing cycle. One advantage of this invention is that more concentrated standardized dose of aerosol is delivered to the patient with the first parcel of gas that enters the lungs for each breath during the inhalation process. In addition, the signal used to actuate the nebulizer may be obtained from the ventilator or from an independently generated signal established by the nebulization system utilizing the readily detected respiratory air line pressure or pressure drop across filter from exhaled gas flow. Also, certain safety monitoring features are incorporated into such a system to detect aerosol clogging of respiratory filters and prevent interference with the normal operation of the respirator.

The nebulization system of the present invention can be attached to or operated with a mechanical respirator utilizing either a breathing cycle electrical signal obtained from the respiratory or an independent electrical signal generated by the nebulizer system which detects and responds to the exhalation initiation of the respirator. Such a synchronized signal actuated nebulizer system is designed to operate during the exhalation phase of the breathing cycle while treating a sick patient and efficiently providing, in the short time available, a medicinal aerosol in the appropriate and desired volume, concentration, and particle size distribution for deposition in the respiratory airways of the lungs. An important feature of such a system is that all of the aerosol is generated quickly (in about 1 second or less) and in a way that does not interfere with the control system of the respirator. The nebulizer system has a reservoir of capacity sufficient to enable several hours of continuous

treatment and with provision to prevent the settling of suspensions or mixtures without creating nebulization-destroying turbulence, and provides a precisely measured volume of medicinal aerosol generated during patient exhalation in a manner to reach the patient at the precise moment when inhalation begins.

In one embodiment, the present invention provides a nebulizer for use with mechanical respirators which use electrical signals to control the breathing cycle. The nebulizer of this embodiment uses the existing electrical signals from the mechanical respirator to synchronize aerosol generation to fill the gas passageway from the respirator to the patient during the exhalation cycle. Upon the initiation of the inhalation cycle, the aerosol is delivered from the gas passageway to the patient. Nebulization is obtained in this embodiment using the premixed oxygen-enriched air provided at high pressure to the respirator. Automatic temperature regulation and stirring of the liquid medication is optionally provided to preclude concentration change, separation or settling of the medication. Finally, a large volume reservoir is provided to eliminate the need for refilling during lengthy treatment protocols.

## Brief Description of the Drawings

Figure 1 is a schematic side view of a nebulizer of the present invention operationally attached to a mechanical respirator;

Figure 2 is a perspective view of the intermittent signal actuated system of the present invention.

Detailed Description of the Invention

Figure 1 shows a nebulizer apparatus 10 of the present invention operably connected to a mechanical respirator 70. The nebulizer apparatus 10 comprises, in a housing, compressed gas inlet 2, at one end of a compressed gas conduit 4, adapted to be connected to a compressed gas source at pressure indicated by gauge 5. Preferably this compressed gas source is the same source which is furnishing oxygen-enriched air to the respirator 70, and provides compressed air or oxygen mixture to the nebulizer ranging up to about 50 psig.

Compressed gas conduit 4 is connected at the other end to a first electrically operated nebulizer valve 7, and a plurality of second electrically operated nebulizer valves 6, all of which are substantially similar. Examples of such valves which have been found useful include the Honeywell Skinner K4M ultraminiature 4-way solenoid operated pneumatic valve and Numatics HS series 2-way solenoid operated valves. Three valves 6 are shown in Figure 1.

Nebulizer valves 6 and 7 are connected by a plurality of electrical lead wires 8 to a microprocessor 9 and are controlled by the microprocessor 9. The microprocessor 9 receives the signals from a signal source 72 on the respirator 70 which controls the inhalation/exhalation phase of the breathing cycle. The microprocessor 9 controls the valves 6 and 7 to provide for a safe and effective operation. Examples of signal source 72 include a respirator solenoid, such as a solenoid actuated inhalation valve, an external electronic monitoring system, or an electronic interface attached to a signal generator on respirator 70, such as an

interface connected to a logic circuit in the respirator.

A control unit 80, whose control panel is shown in Figure 2, is connected to the microprocessor 9. The control unit 80 controls the functions of the nebulizing apparatus 10 of the present invention.

Each of the nebulizer valves 6 connects the compressed gas source 4 to nebulizer conduits 12 leading to aerosol nozzles 22. Each nebulizer valve 6 switches between two positions as electrical on/off signals are received. In the first position, during the exhalation phase of the respirator 70 when the electric signal is "on", a passageway is opened between compressed gas conduit 4 and nebulizer conduits 12 and remain open until the desired aerosol volume has generated or until the inhalation phase is initiated by the respiratory 70 as controlled by microprocessor 9. In the second position, when the electric signal is "off", the nebulizer conduits 12 are sealed off.

Nebulizer conduits 12 are attached at their other ends to aerosol nozzles 22, which include liquid feed tubes 24 extending into reservoir 26. Reservoir 26 includes magnetic stirring bar 28 which is located in the bottom of the reservoir. The liquid medicine contained in reservoir 26 is preferably kept at constant temperature by a reservoir heater or cooler 34.

A chamber 14 houses an AC motor 11 which rotates a cooling fan 13 and a magnet 18. The rotation of the magnet 18 causes the stir bar 28 to rotate to prevent sedimentation or separation of medicinal constituents.

The liquid medicine in the reservoir 26 is drawn via the liquid feed tubes 24 and is converted by the

aerosol nozzles 22 into an aerosol having droplets with a mass median aerodynamic diameter less than about 3 micron. The aerosol is generated into the air space 25 above the reservoir 26. The aerosol generated in the air space 25 enters into an aerosol tube 31.

The temperature of the aerosol in the aerosol tube 31 is controlled by a temperature controller 33. In one embodiment, the temperature controller is simply an electric heater having a control unit. Within the aerosol tube 31 is also a neb-flow sensor 35. The neb-flow sensor 35 detects the amount of aerosol being delivered through the aerosol tube 31. The output of the neb-flow sensor 35 is supplied as a signal to the microprocessor 9 via neb-flow sensor pressure/vacuum lines 17.

The respirator 70 has an inhalation tube 71 and an exhalation tube 73. The inhalation tube 71 fluidically connects the respirator 70 to a patient and during the inhalation phase, breathing gas is supplied from the respirator 70 along the inhalation tube 71 into the respiratory tract of the patient. The aerosol tube 31 connects the air space 25 above the liquid 26 to the inhalation tube 71 at a nebulizer input 30. In addition, a pop-off valve 13 is also located in the inhalation tube 71. The function of the pop-off valve 13 is to relieve any pressure which is generated to dangerous levels within the inhalation tube 71. It functions purely as an emergency safety valve. Finally, an airway pressure sensor 15 is also positioned in the inhalation tube 71. The airway pressure sensor 15 generates a signal which is also supplied to the microprocessor 9 via airway pressure monitoring line 16. A humidifier 37 whose output is water vapor

mixed with the breathing gas is also connected to the inhalation tube 71.

The exhalation tube 73 fluidically connects the patient to the respirator 70. Located within the exhalation tube 73 is an exhalation filter 75. Upstream from the exhalation filter 75, i.e., between the exhalation filter 75 and the patient is an upstream filter pressure sensor 77. Downstream from the exhalation filter 75, i.e., between the exhalation filter 75 and the ventilator 70 is a downstream filter pressure sensor 79. The upstream filter pressure sensor 77 and the downstream filter pressure sensor 79 each provide a signal which is supplied to the microprocessor 9.

The solenoid 7 is also connected to receive gas from the gas conduit 4 and is adapted to supply gas to a decay flow line 11 to the exhalation tube 73, upstream from the upstream filter pressure sensor 77. Thus, the solenoid 7, when activated, provides a stream of compressed gas which is supplied into the exhalation tube 73, between the patient and the upstream filter pressure sensor 77. The function of the decay solenoid 7 is also controlled by the microprocessor 9.

The operation of the nebulizer apparatus 10 of the present invention will be understood as follows. The practitioner first determines the amount of volume per breath of the standardized dose of aerosol which is to be generated by the apparatus 10 of the present invention which is to be supplied to the inhalation tube 71. The amount is entered on the control unit 80. The microprocessor 9 receives the signal and based upon its knowledge of the gas pressure from the compressed gas conduit 4, and the cross-sectional area of each of nebulizing nozzles

22, the microprocessor 9 calculates the amount of time which the solenoids 6 would have to be activated in order to introduce the desired amount of aerosol into the inhalation tube 71. Alternatively, the signal from the neb-flow sensor 35 is used by the microprocessor 9 to turn off the nebulizer solenoids 6 when the desired charging volume has been generated.

When the mechanical respirator 70 begins the exhalation phase of the respiratory cycle, electrical signal 72 supplies the signal to the microprocessor 9. (As will be discussed hereinafter, a number of other signals are supplied to the microprocessor 9 to indicate the beginning of the exhalation cycle. These additional signals are used in the event the ventilator 70 cannot provide the electrical signal source 72 or is used as a safety backup to the electrical signal source 72.) When the mechanical respirator 70 begins the exhalation phase, the inhalation port 76 is closed. The exhalation port 74 is opened, opening the exhalation tube 73.

After the electrical signal source 72 generates the signal indicating the beginning of the exhalation phase, the microprocessor 9 activates the solenoids 6 to the three nebulizing nozzles 24. Thus, after the commencement of the exhalation phase, and after the detection of the electrical signal, maximum generation of the aerosol from the apparatus 10 commences and continues until the standardized volume or dose of aerosol has been generated. Compressed gas flows through the compressed gas conduit 4 into the three nebulizer conduits 12 and into the nozzles 22, which draw liquid via liquid feed tube 24 from the liquid reservoir 26. The aerosol is then generated and is supplied into the air space 25 above

the reservoir 26. The aerosol generated in the air space 25 then enters into the aerosol tube 31 where the temperature thereof is controlled by the temperature controller 33. The aerosol then leaves the aerosol tube 31 and enters into the inhalation tube 71 through port 30. Generation of the standardized dose of aerosol fills the charging volume space 40 between the nebulizer input port 32 and the patient 41 in the inhalation tube 71. Any excessive aerosol will enter the exhalation tube 73 and return to the respirator 70.

During the exhalation phase, the pressure in the inhalation tube 71 is monitored by the airway pressure sensor 15 and is supplied to the microprocessor 9. This provides a safety signal to the microprocessor 9 to shut off the function of the aerosolization in the event pressure within the inhalation tube 71 builds to an excessive level or if inhalation begins. In addition, a mechanical safety pop-off valve 13 is provided wherein in the event the pressure in the inhalation tube 71 exceeds the pressure regulation of the pop-off valve 13, the valve 13 would automatically open relieving the pressure in the inhalation tube 71.

During the exhalation cycle, the respirator 70 continuously monitors the pressure on the exhalation tube 73. In order to provide for a smooth decay flow of gas entering into the exhalation tube 73 from the patient, and thereby simulating smooth exhalation reduction from the patient, the solenoid 7 is activated during the exhalation cycle. When the solenoid 7 is activated, the gas from the compressed gas conduit 4 fills a fixed volume chamber 82. The fixed volume chamber 82 has a calibrated orifice which is connected to the decay flow line 11 and is

supplied to the exhalation tube 73. During the time period in which the aerosol is being generated, the fixed volume chamber 82 is filled with breathing gas to a predetermined pressure. At the end of the charging period, the compressed gas from the gas conduit 4 is turned off. The gas from the fixed volume chamber 82 is then allowed to flow in a decay manner into the exhalation tube through the orifice connecting the chamber 82 to the decay flow line 11. When the pressure in the fixed chamber 82 gradually reduces, the flow entering the decay flow line 11 simulates a natural first order decay.

Synchronous with the beginning of the exhalation cycle, the three nebulizing nozzles 22 are turned on simultaneously or one at a time to produce the desired charging volume during a portion of the exhalation period to allow the respirator 70 to maintain and/or support the patient's spontaneous breathing effort without interference from the charging flow.

When the respirator 70 begins the inhalation phase of the respiratory cycle, the electrical signal source 72 switches to an "off" position. In the "off" position, the respirator inhalation port 76 opens; the respirator exhalation port 74 is closed.

The solenoid valves 6 are controlled by microprocessor 9 when first, the desired standardized dose is reached (usually only takes a portion of the exhalation phase), or secondly when microprocessor 9 detects the electrical signal source 72 turn to an "off" position. In the first priority, the solenoids 6 can be turned off one at a time. In the second case, the solenoids 6 are turned off immediately to allow respirator 70 to begin the inhalation phase.

The gradual turning off of the plurality of solenoids 6 generates a gradual pressure reduction and flow shaping that prevents spurious triggering of the respiratory ventilator 70 caused by rapid flow changes. Because the aerosol generated by the apparatus 10 of the present invention fills the inhalation tube 71 between the nebulizer input 30 and the patient with the desired standardized volume or aerosol dose, when the ventilator 70 begins the inhalation phase and pushes the gas in the inhalation tube 71 into the respiratory tract of the patient, the aerosol in the charging volume space 40 would be the first gas pushed into the lungs of the patient. Thus, the medicine produced by the aerosol would be first delivered to the patient during the inhalation cycle.

The advantage of the apparatus 10 and method of the present invention is that generating the aerosol and introducing it into the charging volume space 40 during the exhalation phase means the aerosol is pre-charged in the inhalation tube. Further, the amount of aerosol in the charging volume space 40 can be metered or controlled by the microprocessor 9. In addition, the introduction of aerosol during the exhalation phase does not perturb the pressure of the gas from the respirator 70 delivered during the inhalation phase.

As previously discussed, the source of electrical signal 72 may not be provided by all ventilators 70. The upstream filter sensor 77 and the downstream filter sensor 79 each provides a signal via the exhalation filter sensor pressure/vacuum lines 19, the difference of which indicates the commencement of the exhalation phase. Thus, upon the immediate commencement of the

exhalation phase, a pressure differential would be detected between the upstream filter sensor 77 and the downstream filter sensor 79, respectively. This pressure differential, supplied as a signal to the microprocessor 9, would indicate to the microprocessor 9 that the exhalation cycle has commenced. This signal can be used by microprocessor 9 to begin nebulization when no respirator electrical signal is available. Alternatively, the airway pressure sensor 15 supplies a signal to the microprocessor 9 indicating the beginning of the exhalation and also the beginning of the inhalation for control of the nebulization by microprocessor 9 when no respirator electrical signal is available.

In addition, there are many safety considerations with the apparatus 10 of the present invention. With the upstream and downstream filter sensor 77 and 79 respectively having an exhalation filter 75 therebetween, the condition of the exhalation filter 75 can be continuously checked. As the apparatus 10 of the present invention is continuously used, and as the filter 75 becomes increasingly clogged, the pressure differential between the upstream filter sensor 77 and the downstream filter sensor 79 would increase. Alternatively, the loading/clogging of the exhalation filter can be detected using the airway pressure sensor 15 which supplies a signal to microprocessor 9 via line 16. This is because airway pressure during nebulization is a function of the resistance of the exhalation filter. The filter loading/clogging can be detected by the microprocessor 9 and can be signaled on the control unit 80 as an alarm that the exhalation filter 75 needs to be examined and/or changed.

As previously discussed, the airway pressure sensor 15 provides an independent airway pressure measurement upstream to exhalation filter to monitor the patients safety. Finally, the control unit 80 can control the apparatus 10 to cause it to pause its operation. This provides an independent check on the respirator system 70. The control unit shown in Figure 2 provides for setting of charging volume, respirator selection (for different commercial respirators), heater temperature, nebulizer hold option, alarm test option, alarm reset, and alarm silence. Further, the control unit displays respirator selection, charging volume, alarm, warning, and caution, indication of exhalation filter loading, patient peak inspiratory pressure, heater temperature and nozzle gas pressure. Signals from the neb-flow sensor 35 are used to alarm if either inadequate charging volume is generated or if the nebulizer nozzle 24 malfunction in the "on" position. The microprocessor 9 provides yet additional safe and effective operation for the apparatus 10 of the present invention. In the preferred embodiment, the microprocessor 9 is an Intel 8751 available from Intel Corporation. A copy of the program, written in the assembly language, for execution by the microprocessor 9 is attached as Exhibit A.

```
                        :VISAN9A
                $TITLE SAMPLE SIGNALS AND CONTROL VISAN 9
                :SAMPLE VENTILATOR ANALOG SIGNAL AND
                :PRESSURE AND FLOW SIGNALS FROM NEBULIZER
                :AND CONTROL 3 NEBULIZER VALVES.
                :CONTROL SERIAL INTERFACE WITH OPERATOR
                :SWITCHES AND DISPLAYS.
                :
0008 =          FLOTIM EQU 11   :TIME=2.2S
0032 =          NOFLOTIM EQU 50 :TIME=10S
002D =          FLO_TH EQU 45   :FLO 18LPM,0.14CMWC,0.17V,2DH
008C =          NOFLO_TH EQU 140 :FLO 35LPM,1.12CMWC,0.5V,8CH
00E0 =          PIP_THRESH SET 120*8/5+32 ;THR=4.4V,E0H,120CM
0037 =          FILTAWP_THRESH EQU 55 :PRES=34CM,1.07V,37H
008D =          FILTDP_THRESH EQU 141 :PRES=5.5CM,2.75V,8DH
0008 =          PATINSP_THRESH SET 5*8/5 ;PEEP-AWP= 5 CM WC
00A0 =          TEMP_HI SET 80*2 ;UPPER LIM 80C,A0H
                :
0000            FSEG
                                :BANK0
0001 =          ALTNAME R1,RVENT_SIG :VENTILATOR SIGNAL
0002 =          ALTNAME R2,RFLT_FLO ;EXH FILT DP SIGNAL
0003 =          ALTNAME R3,RAW_PRESS ;AWP TAP AT VENT
0004 =          ALTNAME R4,RNEB_FLO ;NEB OUTPUT DP
0005 =          ALTNAME R5,RTEMP :TEMP DEG C * 2
0006 =          ALTNAME R6,RVENT  :VENTILATOR # SELECTED
                                :BANK1
0001 =          ALTNAME R1,RCHG_TIM ;NEB CHARGE TIME
0002 =          ALTNAME R2,RDIV10 :TIMER DIV BY 10
0003 =          ALTNAME R3,RDIV5  ;TIMER DIV BY 5
0004 =          ALTNAME R4,RON_TIM ;NEB FLOW ON TIME
0005 =          ALTNAME R5,ROFF_TIM ;NEB FLOW OFF TIME
0006 =          ALTNAME R6,RSIL_TIM ;AUDIO OFF TIME
0007 =          ALTNAME R7,RHOLD_TIM ;NEB OFF TIME
0000            ENDS
                :
0000            DSEG
0023 =          LED1 DATA 23H :LED BANKS
0026 =          LED2 DATA 26H
0025 =          LED3 DATA 25H
0028 =          CHG_VOL DATA 28H :HUNS DEC DISPLAY
0029 =          DEC_HUN DATA 29H :NUMBER FOR DISPLAY
002A =          DEC_TEN DATA 2AH
002B =          DEC_ONE DATA 2BH
002C =          FLTLD_HUN DATA 2CH :FILTER LOAD SETTING
002D =          FLTLD_TEN DATA 2DH : 25%, 50% OR 75%
002E =          FLTLD_ONE DATA 2EH
002F =          THREE_CYCLE DATA 2FH :THREE BREATH COUNTS
0040 =          FLTFLO_LO DATA 40H :RUNNING AVG CALC
0044 =          FLTFLO_AVG DATA 44H
0045 =          CLOG_LO DATA 45H
0046 =          CLOG_HI DATA 46H
0048 =          AWP_LO DATA 48H
004C =          AWP_AVG DATA 4CH
004D =          AWP_MAX DATA 4DH
```

```
004E =          PIP_STORE DATA 4EH
0050 =          POSSUM DATA 50H ;NEB POS SUM
0051 =          NEGSUM DATA 51H ;NEB NEG SUM
0055 =          FLTLD25 DATA 55H :PERCENT FILTER LOAD
0056 =          FLTLD50 DATA 56H
0057 =          FLTLD75 DATA 57H
0058 =          PIP_LO DATA 58H
0058 =          PIP_AVG DATA 5BH
0060 =          PEEP_LO DATA 60H
0063 =          PEEP_AVG DATA 63H
0011 =          TEMP_SET DATA 11H
0012 =          TEMP_DEC DATA 12H
0014 =          ONTIMER DATA 14H
0015 =          OFFTIMER DATA 15H
0019 =          SET_CHGTIM DATA 19H :CONTROLS CHARGE VOL
001A =          VENT_LO DATA 1AH     :LOWER THRESH
001B =          VENT_HI DATA 1BH     ;UPPER THRESH
001C =          TEMP_STORE DATA 1CH :TEMPORARY STORE
001D =          DIVIDE1 DATA 1DH   :TRANS_DEL
001E =          DIVIDE2 DATA 1EH
0068 =          VENT_LOW DATA 68H
006C =          VENT_AVG DATA 6CH
0028            ENDS
                ;
0000            BSEG
0000 =          WAIT BIT 0H :FIVE BREATH WAIT
0001 =          EXH BIT 1H :EXHALATION PERIOD
0002 =          DIV21 BIT 2H :TIMER
0003 =          VOL_CHG BIT 3H :OP CHANGING VOL SET
0004 =          VEN_SEL BIT 4H :OP SELECTING VENTILATOR
0014 =          BEEP BIT 14H :AUDIO ON/OFF
0006 =          SIL BIT 6H   ;TWO MIN SILENCE
0007 =          SPON_BR BIT 7H ;PATIENT BREATH
0008 =          HOLD BIT 8H   ;NEB OFF
0009 =          SEE_PIP BIT 09H :DISPLAY PIP
0008 =          DIV22 BIT 0BH   :TIMER
000C =          ALM BIT 0CH :AUDIO ALM SET
000D =          OFF_ALM BIT 0DH ;BLINK_BEEP
000A =          ALM_TST BIT 0AH :SET DURING TEST
000E =          DIV24 BIT 0EH   ;START DELAY
000F =          FLOW BIT 0FH :NEB FLOW ON
0010 =          SEE_TEMP BIT 10H
0011 =          SEE_LD BIT 11H
0012 =          DEL1 BIT 12H
0013 =          DEL_4TENTHS BIT 13H   :TIMER
0015 =          INSP BIT 15H :INSP TIME
0016 =          CLOG1 BIT 16H :COUNT FLT LD SAMP
0017 =          CLOG2 BIT 17H
001C =          L14 BIT 1CH :LO BAT          LED1
001D =          L15 BIT 1DH :FILTER CHANGE
001E =          L16 BIT 1EH :WAIT 5 CYCLES
001F =          L17 BIT 1FH ;LO FLOW
0034 =          L24 BIT 34H :NO FLOW         LED2
0035 =          L25 BIT 35H :NEB HOLD
0036 =          L26 BIT 36H :FILT CLOG
```

```
0037 =          L27 BIT 37H :CONT FLOW
002C =          L34 BIT 2CH :HI PRESS       LED3
0020 =          L35 BIT 20H :HI TEMP
0038 =          DIV23 BIT 38H :TIMER
0039 =          CLK BIT 39H    :TIMER 0.2S
003A =          HEAT BIT 3AH   :HEATER ON
003B =          TEMP BIT 3BH
0025            ENDS
                ;
0000            CSEG
                ; MACRO DEFINITIONS
                ;
                ANALOG MACRO SAVE      :ANALOG-DIGITAL CONVERSION
                NOP   :DELAY TIME FOR MUX
                NOP
                NOP
                NOP
                CLR P2.3       :START CONVERSION
                NOP      ;ALLOW CONV. TIME 5 MICROSEC
                NOP
                NOP
                MOV SAVE,P1      ;SAVE DIGITAL OUTPUT
                SETB P2.3
                ENDM
                ;
                ;
                RUNNING_AVG MACRO LODATA,N,INSIG,AVG
                ;CALCULATES RUNNING AVERAGE OF N BYTES IN DATA MEMORY
                ;WITH A LOW ADDRESS OF LODATA. INPUT SIGNAL IS LOCATED
                ;AT INSIG. AVERAGE OUTPUT IS AT AVG.
                PUSH PSW
                PUSH ACC
                PUSH B
                CLR PSW.3   :BANK0
                CLR PSW.4
                MOV A,#LODATA    :SET R0
                ADD A,#N
                DEC A
                MOV R0,A
                NEXT1:
                DEC R0
                MOV A,@R0     :SHIFT UP
                INC R0
                MOV @R0,A
                DEC R0
                CJNE R0,#LODATA,NEXT1   :LODATA ADDRESS
                MOV A,INSIG     :MOV NEW DATA TO LODATA
                MOV B,#N
                DIV AB
                MOV @R0,A
                MOV A,#LODATA   :ADD TO CALC AVG
                ADD A,#N
                DEC A
                MOV TEMP_STORE,A
```

```
MOV A,@R0
XCH A,R0
NEXT2:
XCH A,R0
INC R0
ADD A,@R0
XCH A,R0
CJNE A,TEMP_STORE,NEXT2
XCH A,R0
MOV AVG,A
POP B
POP ACC
POP PSW
ENDM

FIFO MACRO NEW_IN,N1,NEW_DATA
;REGISTER STORES SUCCESSIVE DATA.FIFO
;FROM NEW_DATA SOURCE INTO REGISTER ADDRESS
;NEW_IN. N1 IS THE NUMBER OF DATA STORED.
CLR PSW.3    ;BANK0
CLR PSW.4
MOV A,#NEW_IN   ;SET R0
ADD A,#N1
DEC A
MOV R0,A
NEXT3:
DEC R0
MOV A,@R0   ;SHIFT UP
INC R0
MOV @R0,A
DEC R0
CJNE R0,#NEW_IN,NEXT3   ;NEW_IN ADDR
MOV NEW_IN,NEW_DATA
ENDM
BINARY_BCD MACRO HUN,TEN,ONE
:CONVERTS BYTE LOCATED IN ACC TO  DECIMAL
;AND STORES RESULT IN HUN, TEN AND ONE.
MOV HUN,#0   ;CLEAR REGISTERS
MOV TEN,#0
MOV ONE,#0
CALC_HUN:   ::SUBTRACT 100
MOV B,A
NEXTSUB1:
CLR C
SUBB A,#100
JC CALC_TEN
INC HUN
MOV B,A     ;SAVE
SJMP NEXTSUB1
CALC_TEN:   ::SUBTRACT 10
MOV A,B
NEXTSUB2:
CLR C
SUBB A,#10
JC CALC_ONE
```

```
                INC TEN
                MOV B,A
                SJMP NEXTSUB2
                CALC_ONE:
                MOV ONE,B
                MOV A,HUN
                JNZ BCD_OUT
                MOV HUN,#0FH   :BLANK
                MOV A,TEN
                JNZ BCD_OUT
                MOV TEN,#0FH   :BLANK
                BCD_OUT:
                ENDM
                :
                :
1000            ORG 1000H
                :%S
                BEGIN:
1000 0130       AJMP INITIALIZE
1003            ORG 1003H :MANUAL SWITCH INT..INTO
1003 0219CC     LJMP MAN_SW
1008            ORG 1008H :TIMER 0 INT..TF0
1008 61F3       AJMP TIM_SAMP
1013            ORG 1013H :LOW BATTERY INT..INT1
1013 C28B       CLR IE1
1015 D21C       SETB L14
1017 852399     MOV SBUF,LED1
101A D125       ACALL TRANS_DEL
101C 32         RETI
                :
1030            ORG 1030H
                INITIALIZE:  ::SET REGISTERS
1030 D212       SETB DEL1
                INIT1:
1032 538700     ANL PCON,#00H : SMOD = 0
1035 758920     MOV TMOD,#00100000B ; TIME 1 MODE 2, TIME 0 MODE 0
1038 759850     MOV SCON,#01010000B : SERIAL PORT MODE 1
103B 758C70     MOV TH0,#70H   :SET TIMER
103E 758DFD     MOV TH1,#0FDH  : BAUD RATE 9600
1041 75A078     MOV P2,#78H    ; OUTPUTS OFF
1044 75A887     MOV IE,#87H    :ENABLE EX1,ET0,EX0
1047 758802     MOV IP,#02H    :FIRST PRIORITY TIMER 0
104A 758850     MOV TCON,#50H  :TIMERS ACTIVE, IT1 & IT0
                               :LOW LEVEL TRIGGGER
104D 758000     MOV P0,#00H
1050 758130     MOV SP,#30H  : STACK ADDRESS
1053 752000     MOV 20H,#00H :CLEAR BITS
1056 752100     MOV 21H,#00H
1059 752200     MOV 22H,#00H
105C 752700     MOV 27H,#00H
105F D2D3       SETB PSW.3      :BANK1
1061 7B05       MOV RDIV5,#5    :R3
1063 7A0A       MOV RDIV10,#10  :R2
1065 7E78       MOV RSIL_TIM,#120  :R6,DEL 2 MIN (3CH)
1067 7F78       MOV RHOLD_TIM,#120 :R7
```

```
1069 7D32       MOV ROFF_TIM,#NOFLOTIM ;R5. CLEAR REGISTER
1068 7C08       MOV RON_TIM,#FLOTIM ;R4
106D 7900       MOV RCHG_TIM,#0 ;R1
106F 751532     MOV OFFTIMER,#NOFLOTIM
1072 751408     MOV ONTIMER,#FLOTIM
1075 755000     MOV POSSUM,#0
1078 755100     MOV NEGSUM,#0
1078 751128     MOV TEMP_SET,#40  ;DEFAULT
107E 752C00     MOV FLTLD_HUN,#00H
1081 752D01     MOV FLTLD_TEN,#01H
1084 752E02     MOV FLTLD_ONE,#02H
1087 751DFF     MOV DIVIDE1,#0FFH    ;TRANS DEL
108A 751E04     MOV DIVIDE2,#04H
108D D202       SETB DIV21  ;TIMER
108F D208       SETB DIV22
1091 751B45     MOV VENT_HI,#45H  ;THRESH = 2.7V/2 = 1.35V
1094 751A38     MOV VENT_LO,#38H  ;THRESH = 2.3V/2 = 1.15V
1097 751928     MOV SET_CHGTIM,#40 ;CASE8 GIVES 60
109A C2D3       CLR PSW.3         ;BANK0
109C 7E13       MOV RVENT,#13H   ;R6,VENT #
109E 8E99       MOV SBUF,RVENT
10A0 D125       ACALL TRANS_DEL
10A2 752344     MOV LED1,#44H    ;WAIT LED ON
10A5 852399     MOV SBUF,LED1
10A8 D125       ACALL TRANS_DEL
10AA 752605     MOV LED2,#05H
10AD 852699     MOV SBUF,LED2
10B0 D125       ACALL TRANS_DEL
10B2 752506     MOV LED3,#06H
10B5 852599     MOV SBUF,LED3
10B8 D125 .     ACALL TRANS_DEL
10BA 752840     MOV CHG_VOL,#40H  ;CASE8 GIVES 600ML
10BD 301212     JNB DEL1,CONT5
10C0 C212       CLR DEL1
10C2 C208       CLR DIV22
10C4 C20E       CLR DIV24
10C6 200E02     DELAY1: JB DIV24,DELAY2
10C9 80FB       SJMP DELAY1
10CB 300E02     DELAY2: JNB DIV24,END_DEL
10CE 80FB       SJMP DELAY2
10D0 0132       END_DEL: AJMP INIT1
10D2 121765     CONT5: LCALL CASE81
10D5 00         NOP
10D6 00         NOP
10D7 00         NOP
                ;
                MAIN_LOOP:   ;:INSP/EXP CYCLE
10D8 12156E     LCALL SERVICE
10DB 200C51     JB ALM,ALARM
10DE C2D3       CLR PSW.3 ;BANK0
10E0 C2D4       CLR PSW.4
10E2 E518       MOV A,VENT_HI ;WAIT FOR SOI
10E4 C3         CLR C
10E5 956C       SUBB A,VENT_AVG ;R1
10E7 50EF       JNC MAIN_LOOP  ;?NOT INSP
```

```
10E9 D215       SETB INSP

                EOI: ::WAIT FOR EOI
10EB 12156E     LCALL SERVICE
10EE 200C3E     JB ALM.ALARM
10F1 E51A       MOV A.VENT_LO
10F3 C3         CLR C
10F4 956C       SUBB A.VENT_AVG :R1
10F6 40F3       JC EOI    :?NOT EOI
10F8 C215       CLR INSP

10FA D2D3       SETB PSW.3   :BANK1
10FC C2D4       CLR PSW.4
10FE 7900       MOV RCHG_TIM.#00H :R1
                CHK_EXH:        ::FIND AWP PEAK & DROP
1100 E54D       MOV A.AWP_MAX
1102 C3         CLR C
1103 954C       SUBB A.AWP_AVG
1105 4018       JC DELAY5       :?AWP MAX > AWP AVG
                CHK_AWP:       ::CHK AWP DROP
1107 F5F0       MOV B.A    :SAVE
1109 E54D       MOV A.AWP_MAX
110B C3         CLR C
110C 9563       SUBB A.PEEP_AVG :AWP MAX - PEEP
110E 4007       JC SET_EXH   :AWP<PEEP
1110 84         DIV AB
1111 9405       SUBB A.#5
1113 4002       JC SET_EXH   :?DROP 20%
1115 2122       AJMP DELAY5
1117 D201       SET_EXH:  SETB EXH
1119 854D4E     MOV PIP_STORE.AWP_MAX  :NEW PIP
111C 754D00     MOV AWP_MAX.#0  ;RESET
111F 2150       AJMP CHK_PEAK

1121 00         NOP
                DELAY5:  ::WAIT 0.5S
1122 D2D3       SETB PSW.3   :BANK1
1124 C2D4       CLR PSW.4
1126 7432       MOV A.#50
1128 C3         CLR C
1129 99         SUBB A.RCHG_TIM
112A 50D4       JNC CHK_EXH :?NOT 0.5S
112C 00         NOP
112D 00         NOP
112E 00         NOP

                ALARM:
112F D20C       SETB ALM
1131 43A070     ORL P2.#01110000B   :OFF VALVES
1134 200605     CHK_SIL: JB SIL.CONT
1137 200802     JB HOLD.CONT
113A D2A7       SETB P2.7 :BUZZER ON
113C D200       CONT: SETB WAIT
113E D21E       SETB LI6   :WAIT
1140 8523B0     MOV SBUF.LED1
```

```
1143 D125       ACALL TRANS_DEL
1145 12156E     LCALL SERVICE
1148 200CE9     JB ALM.CHK_SIL
1148 752F00     MOV THREE_CYCLE.#0
114E 01D8       AJMP MAIN_LOOP

                CHK_PEAK:    ::PRESS LIMIT 120 CM
1150 20000A     JB WAIT.CALC_PIP
1153 E54E       MOV A.PIP_STORE
1155 C3         CLR C
1156 9563       SUBB A.PEEP_AVG
1158 C3         CLR C
1159 94E0       SUBB A.#PIP_THRESH
115B 5046       JNC HIPRESS
                CALC_PIP:
115D            RUNNING_AVG PIP_LO.3.PIP_STORE.PIP_AVG
+               :CALCULATES RUNNING AVERAGE OF 3 BYTES IN DATA MEMORY
+               :WITH A LOW ADDRESS OF LODATA. INPUT SIGNAL IS LOCATED
+               :AT INSIG. AVERAGE OUTPUT IS AT AVG.
+115D C0D0       PUSH PSW
+115F C0E0       PUSH ACC
+1161 C0F0       PUSH B
+1163 C2D3       CLR PSW.3   :BANK0
+1165 C2D4       CLR PSW.4
+1167 7458       MOV A.#PIP_LO    :SET R0
+1169 2403       ADD A.#3
+116B 14         DEC A
+116C F8         MOV R0.A
+               NEXT10001:
+116D 18         DEC R0
+116E E6         MOV A.@R0     :SHIFT UP
+116F 08         INC R0
+1170 F6         MOV @R0.A
+1171 18         DEC R0
+1172 8858F8     CJNE R0.#PIP_LO.NEXT10001   :LODATA ADDRESS
+1175 E54E       MOV A.PIP_STORE     :MOV NEW DATA TO PIP_LO
+1177 75F003     MOV B.#3
+117A 84         DIV AB
+117B F6         MOV @R0.A
+117C 7458       MOV A.#PIP_LO   :ADD TO CALC PIP_AVG
+117E 2403       ADD A.#3
+1180 14         DEC A
+1181 F51C       MOV TEMP_STORE.A
+1183 E6         MOV A.@R0
+1184 C8         XCH A.R0
+               NEXT20001:
+1185 C8         XCH A.R0
+1186 08         INC R0
+1187 26         ADD A.@R0
+1188 C8         XCH A.R0
+1189 851CF9     CJNE A.TEMP_STORE.NEXT20001
+118C C8         XCH A.R0
+118D F55B       MOV PIP_AVG.A
+118F D0F0       POP B
+1191 D0E0       POP ACC
```

```
+1193 D0D0      POP PSW
 1195 00        NOP
 1196 00        NOP
 1197 00        NOP
 1198 200013    JB WAIT.STRT_EXH
 1198 200810    JB HOLD.STRT_EXH
 119E 53A08F    ANL P2.#10001111B    ;ON VALVES
 11A1 21AE      AJMP STRT_EXH
                HIPRESS:
 11A3 D22C      SETB L34  :HI PRESS
 11A5 852599    MOV SBUF.LED3
 11A8 D125      ACALL TRANS_DEL
 11AA 00        NOP
 11A8 212F      ALARM1: AJMP ALARM
 11AD 00        NOP

                STRT_EXH:
 11AE D2D3      SETB PSW.3           :BANK1
 11B0 C2D4      CLR PSW.4
 11B2 7900      MOV RCHG_TIM.#00H    :R1.RST CHARGE TIME

                CHARGE:
 11B4 C2D3      CLR PSW.3              :BANK 0
 11B6 12156E    LCALL SERVICE
 11B9 200CEF    JB ALM.ALARM1
 11BC E518      MOV A.VENT_HI     :VENTILATOR INSPIRATION?
 11BE C3        CLR C
 11BF 956C      SUBB A.VENT_AVG
 11C1 5023      JNC CHK_CHGTIM      ;?NO VENT INSP1
 11C3 43A070    ORL P2,#01110000B   ;OFF VALVES

 11C6 D2D3      CHK_VOL: SETB PSW.3   :BANK1
 11C8 C2D4      CLR PSW.4
 11CA 200016    JB WAIT,CHK_WAIT1
 11CD E519      MOV A.SET_CHGTIM
 11CF C3        CLR C
 11D0 99        SUBB A.RCHG_TIM   :R1
 11D1 4010      JC CHK_WAIT1   ::VOL>SET
 11D3 F5F0      MOV B.A
 11D5 E519      MOV A,SET_CHGTIM
 11D7 84        DIV AB
 11D8 940A      SUBB A.#10
 11DA 5007      JNC CHK_WAIT1
 11DC D21F      SETB L17  :LO FLOW LED
 11DE 852399    MOV SBUF.LED1
 11E1 D125      ACALL TRANS_DEL
 11E3 6108      CHK_WAIT1:  AJMP CHK_WAIT
 11E5 00        NOP
                CHK_CHGTIM:
 11E6 E519      MOV A.SET_CHGTIM  :SET VOLUME REACHED?
 11EB D2D3      SETB PSW.3          :BANK1
 11EA C3        CLR C
 11EB 99        SUBB A.RCHG_TIM :R1
 11EC 50C6      JNC CHARGE   :?VOL < SET VOL
 11EE 43A070    ORL P2.#01110000B      :OFF VALVES
```

```
11F1 20004B     JB WAIT.CHK_EOEXH1

11F4 101629     JBC CLOG1.FIRST_SAMP :MEAS FLT LD SAMP
11F7 30172B     JNB CLOG2.FLT_LD
11FA C217       CLR CLOG2  :SECOND SAMPLE
11FC E544       MOV A.FLTFLO_AVG
11FE 2545       ADD A.CLOG_LO
1200 F546       MOV CLOG_HI.A  :UPPER LIM FILT CLOG
1202 C3         CLR C
1203 13         RRC A  :DIV BY 2
1204 F545       MOV CLOG_LO.A  :LOWER LIM FILT CLOG
1206 C3         CLR C
1207 13         RRC A  ;HALF CLOG LO
1208 F5F0       MOV B.A  :SAVE
120A 2545       ADD A.CLOG_LO
120C F556       MOV FLTLD50.A  :STORE 50% LEVEL
120E E5F0       MOV A.B
1210 C3         CLR C
1211 13         RRC A  :ONE FOURTH CLOG LO
1212 F5F0       MOV B.A  :SAVE
1214 2545       ADD A.CLOG_LO
1216 F555       MOV FLTLD25.A  :STORE 25% LEVEL
1218 E5F0       MOV A.B
121A 2556       ADD A.FLTLD50
121C F557       MOV FLTLD75.A  :STORE 75% LEVEL
121E 4142       AJMP CHK_DPTHRESH

                FIRST_SAMP:    ::FIRST FLT LD SAMP
1220 854445     MOV CLOG_LO.FLTFLO_AVG  :SAVE
1223 4142.      AJMP CHK_DPTHRESH

                FLT_LD: ::SAVE FILT LOAD %
1225 E544       MOV A.FLTFLO_AVG
1227 C3         CLR C
1228 9546       SUBB A.CLOG_HI
122A 402F       JC TEST75
122C D236       SETB L26  :FILTER CLOG LED
122E 852399     MOV SBUF.LED1
1231 D125       ACALL TRANS_DEL
1233 752C10     MOV FLTLD_HUN.#10H  :SET FILTER LOAD 100%
1236 752D01     MOV FLTLD_TEN.#01H
1239 752E02     MOV FLTLD_ONE.#02H
123C 212F       AJMP ALARM

123E 00         NOP
123F 41A7       CHK_EOEXH1: AJMP CHK_EOEXH
1241 00         NOP

                CHK_DPTHRESH:
1242 E544       MOV A.FLTFLO_AVG
1244 C3         CLR C
1245 948D       SUBB A.#FILTDP_THRESH
1247 40F6       JC CHK_EOEXH1  :BELOW THRESH
1249 D236       SETB L26  :FILT CLOG LED
124B 852600     MOV SBUF.LED2
```

```
124E D125       ACALL TRANS_DEL
1250 752C10     MOV FLTLD_HUN.#10H
1253 752D01     MOV FLTLD_TEN.#01H
1256 752E02     MOV FLTLD_ONE.#02H
1259 212F       AJMP ALARM

                TEST75:    ::TEST 75% CLOG
125B E544       MOV A.FLTFLO_AVG
125D C3         CLR C
125E 9557       SUBB A.FLTLD75
1260 4012       JC TEST50
1262 D21D       SETB L15    :FILTER CHANGE LED
1264 852399     MOV SBUF.LED1
1267 D125       ACALL TRANS_DEL
1269 752CF0     MOV FLTLD_HUN.#0F0H   :BLANK
126C 752D71     MOV FLTLD_TEN.#71H
126F 752E52     MOV FLTLD_ONE.#52H
1272 4142       AJMP CHK_DPTHRESH
                TEST50:   ::TEST 50% CLOG
1274 E544       MOV A.FLTFLO_AVG
1276 C3         CLR C
1277 9556       SUBB A.FLTLD50
1279 4008       JC TEST25
127B 752CF0     MOV FLTLD_HUN.#0F0H
127E 752D51     MOV FLTLD_TEN.#51H
1281 752E02     MOV FLTLD_ONE.#02H
1284 4142       AJMP CHK_DPTHRESH
                TEST25:    ::TEST 25% CLOG
1286 E544       MOV A.FLTFLO_AVG
1288 C3         CLR C
1289 9555       SUBB A,FLTLD25
128B 4008       JC TEST0
128D 752CF0     MOV FLTLD_HUN.#0F0H
1290 752D21     MOV FLTLD_TEN.#21H
1293 752E52     MOV FLTLD_ONE.#52H
1296 4142       AJMP CHK_DPTHRESH
                TEST0:
1298 752CF0     MOV FLTLD_HUN.#0F0H
1298 752DF1     MOV FLTLD_TEN.#0F1H
129E 752E02     MOV FLTLD_ONE.#02H
12A1 4142       AJMP CHK_DPTHRESH
12A3 00         NOP

12A4 212F       ALARM2: AJMP ALARM
12A6 00         NOP
                CHK_EOEXH:
12A7 12156E     LCALL SERVICE
12AA 200CF7     JB ALM.ALARM2
12AD C2D3       CLR PSW.3   :BANK0
12AF C2D4       CLR PSW.4
12B1 E518       MOV A.VENT_HI
12B3 956C       SUBB A.VENT_AVG   :R1
12B5 503F       JNC PAT_INSP
12B7 C201       CLR EXH   :END OF EXHALATION
12B9            RUNNING_AVG PEEP_LO.3.AWP_AVG.PEEP_AVG
```

```
+                    :CALCULATES RUNNING AVERAGE OF 3 BYTES IN DATA MEMORY
+                    :WITH A LOW ADDRESS OF LODATA. INPUT SIGNAL IS LOCATED
+                    :AT INSIG. AVERAGE OUTPUT IS AT AVG.
+12B9 C0D0           PUSH PSW
+12BB C0E0           PUSH ACC
+12BD C0F0           PUSH B
+12BF C2D3           CLR PSW.3   :BANK0
+12C1 C2D4           CLR PSW.4
+12C3 7460           MOV A,#PEEP_LO    :SET R0
+12C5 2403           ADD A,#3
+12C7 14             DEC A
+12C8 F8             MOV R0,A
+                    NEXT10002:
+12C9 18             DEC R0
+12CA E6             MOV A,@R0     :SHIFT UP
+12CB 08             INC R0
+12CC F6             MOV @R0,A
+12CD 18             DEC R0
+12CE B860F8         CJNE R0,#PEEP_LO,NEXT10002  :LODATA ADDRESS
+12D1 E54C           MOV A,AWP_AVG     :MOV NEW DATA TO PEEP_LO
+12D3 75F003         MOV B,#3
+12D6 84             DIV AB
+12D7 F6             MOV @R0,A
+12D8 7460           MOV A,#PEEP_LO  :ADD TO CALC PEEP_AVG
+12DA 2403           ADD A,#3
+12DC 14             DEC A
+12DD F51C           MOV TEMP_STORE,A
+12DF E6             MOV A,@R0
+12E0 C8             XCH A,R0
+                    NEXT20002:
+12E1 C8             XCH A,R0
+12E2 08             INC R0
+12E3 26             ADD A,@R0
+12E4 C8             XCH A,R0
+12E5 B51CF9         CJNE A,TEMP_STORE,NEXT20002
+12E8 C8             XCH A,R0
+12E9 F563           MOV PEEP_AVG,A
+12EB D0F0           POP B
+12ED D0E0           POP ACC
+12EF D0D0           POP PSW
 12F1 00             NOP
 12F2 00             NOP
 12F3 00             NOP
 12F4 6108           AJMP CHK_WAIT


                     PAT_INSP:
 12F6 E563           MOV A,PEEP_AVG
 12F8 C3             CLR C
 12F9 954C           SUBB A,AWP_AVG   :PEEP - AWP
 12FB 40AA           JC CHK_EOEXH     :AWP > PEEP
 12FD 9408           SUBB A,#PATINSP_THRESH
 12FF 40A6           JC CHK_EOEXH    :?NO PAT INSP
 1301 C201           CLR EXH
 1303 D207           SETB SPON_BR
```

```
1305 00          NOP
1306 00          NOP
1307 00          NOP

                 CHK_WAIT:  ::CHECK 3 CYC WAIT
1308 30002E      JNB WAIT.GO_ON
130B 200428      JB VEN_SEL.GO_ON
130E 200328      JB VOL_CHG.GO_ON
1311 200825      JB HOLD.GO_ON
1314 7402        MOV A.#2
1316 C3          CLR C
1317 952F        SUBB A.THREE_CYCLE
1319 5018        JNC INC3
131B C200        CLR WAIT
131D C21E        CLR L16
131F D216        SETB CLOG1
1321 D217        SETB CLOG2
1323 8523??      MOV SBUF.LED1
1326 0125        ACALL TRANS_DEL
1328 7C00        MOV RON_TIM.#0   :RESET AFTER WAIT
132A 7D00        MOV ROFF_TIM.#0
132C 755000      MOV POSSUM.#0
132F 755100      MOV NEGSUM.#0
1332 6139        AJMP GO_ON
1334 00          NOP
1335 00          NOP

                 INC3:
1336 052F        INC THREE_CYCLE
1338 00          NOP
                 GO_ON: ::START MAIN LOOP
1339 01D8        AJMP MAIN_LOOP
133B 00          NOP

133C 61EB        OUT1: AJMP OUT
133E 00          NOP

                 BLINK_BEEP:  ::ON/OFF DISPLAY & BUZZER
133F 1038FA      JBC DIV23.OUT1  :PERIOD 0.4S
1342 D238        SETB DIV23
1344 200AF5      JB ALM_TST.OUT1
1347 C2D3        CLR PSW.3    :BANK0
1349 C2D4        CLR PSW.4
134B 100D50      JBC OFF_ALM.TURN_OFF
                 TURN_ON:  ::DISPLAY/ALM ON
134E D20D        SETB OFF_ALM
1350 301F05      JNB L17.CHK_LED21
1353 8523??      MOV SBUF.LED1    :RESTORE LED'S
1356 0125        ACALL TRANS_DEL
                 CHK_LED21:
1358 E526        MOV A.LED2
135A 54F0        ANL A.#0F0H
135C 6005        JZ CHK_LED31
135E 8526??      MOV SBUF.LED2
1361 0125        ACALL TRANS_DEL
```

```
                CHK_LED31:
1363 E525       MOV A,LED3
1365 54F0       ANL A,#0F0H
1367 6005       JZ CHK_VOL1
1369 8525??     MOV SBUF,LED3
136C D125       ACALL TRANS_DEL
136E 30030F     CHK_VOL1:  JNB VOL_CHG,TST_VENTSEL1
1371 8528??     MOV SBUF,CHG_VOL  ;SET HUNS
1374 D125       ACALL TRANS_DEL
1376 759901     MOV SBUF,#01H     ;SET TENS TO 0
1379 D125       ACALL TRANS_DEL
137B 759902     MOV SBUF,#02H     ;SET ONES TO 0
137E D125       ACALL TRANS_DEL
                TST_VENTSEL1:
1380 300404     JNB VEN_SEL,TST_TEMP1
1383 8E99       MOV SBUF,RVENT
1385 D125       ACALL TRANS_DEL
                TST_TEMP1:
1387 303B0A     JNB TEMP,TST_BEEP1
138A 851299     MOV SBUF,TEMP_DEC   ;TENS
138D D125       ACALL TRANS_DEL
138F 759902     MOV SBUF,#02H ;ONES
1392 D125       ACALL TRANS_DEL
                TST_BEEP1:
1394 301454     JNB BEEP,OUT
1397 200651     JB SIL,OUT
139A D2A7       SETB P2.7     ;BUZZER ON
139C 61EB       AJMP OUT
                TURN_OFF:     ::DISPLAY/ALM OFF
139E 301F08     JNB L17,CHK_LED22
13A1 E523       MOV A,LED1
13A3 547F       ANL A,#7FH    ;MASK LED'S
13A5 F599       MOV SBUF,A
13A7 D125       ACALL TRANS_DEL
                CHK_LED22:
13A9 E526       MOV A,LED2
13AB 54F0       ANL A,#0F0H
13AD 6005       JZ CHK_LED32
13AF 759905     MOV SBUF,#05H
13B2 D125       ACALL TRANS_DEL
                CHK_LED32:
13B4 E525       MOV A,LED3
13B6 54F0       ANL A,#0F0H
13B8 6005       JZ CHK_VOL2
13BA 759906     MOV SBUF,#06H
13BD D125       ACALL TRANS_DEL
13BF 30030F     CHK_VOL2: JNB VOL_CHG,TST_VENTSEL2
13C2 7599F0     MOV SBUF,#0F0H    ;OFF HUNS
13C5 D125       ACALL TRANS_DEL
13C7 7599F1     MOV SBUF,#0F1H    ;OFF TENS
13CA D125       ACALL TRANS_DEL
13CC 7599F2     MOV SBUF,#0F2H    ;OFF ONES
13CF D125       ACALL TRANS_DEL
                TST_VENTSEL2:
13D1 300405     JNB VEN_SEL,TST_TEMP2
```

```
13D4 7599F3     MOV SBUF,#0F3H     ;VENT SEL OFF
13D7 D125       ACALL TRANS_DEL
                TST_TEMP2:
13D9 303B0A     JNB TEMP.TST_BEEP2
13DC 7599F1     MOV SBUF,#0F1H   ;OFF TENS
13DF D125       ACALL TRANS_DEL
13E1 7599F2     MOV SBUF,#0F2H   ;OFF ONES
13E4 D125       ACALL TRANS_DEL

                TST_BEEP2:
13E6 301402     JNB BEEP.OUT
13E9 C2A7       CLR P2.7    ;AUDIO OFF
                OUT:
13EB 758C70     MOV TH0,#70H   ;RST TIMER0
13EE D2A9       SETB ET0
13F0 D28C       SETB TR0
13F2 22         RET


                TIM_SAMP:      ;;TIMER 0 INTERRUPT
13F3 C0E0       PUSH ACC    ;SAVE SFR'S
13F5 C0F0       PUSH B
13F7 C0D0       PUSH PSW
13F9 758C70     MOV TH0,#70H   ;RESET TIMER
13FC D2D3       SETB PSW.3    ;SELECT REGISTER BANK 1
13FE C2D4       CLR PSW.4
1400 100204     JBC DIV21,CLEAR
1403 D202       SETB DIV21 ;FREQ 100HZ
1405 A167       AJMP RETURN
1407 09         CLEAR:  INC RCHG_TIM ;R1
1408 DA2B       DJNZ RDIV10,SAMPLE ;R2
140A 7A0A       MOV RDIV10,#10 ;RESET RDIV10
140C 100804     JBC DIV22,SET_CLK
140F D208       SETB DIV22
1411 8135       AJMP SAMPLE

                SET_CLK:  ;;SET .2S CLOCK
1413 D239       SETB CLK
1415 100E02     JBC DIV24,CONT6
1418 D20E       SETB DIV24
141A DB19       CONT6:  DJNZ RDIV5,SAMPLE   ;R3
141C 7B05       MOV RDIV5,#5   ;FREQ 1 HZ

                ;SILENCE 2 MIN
141E 300608     JNB SIL,CHK_HOLD
1421 C2A7       CLR P2.7              ;BUZZER OFF
1423 DE04       DJNZ RSIL_TIM,CHK_HOLD     ;?NOT 2 MIN
1425 7E78       MOV RSIL_TIM,#120 ;R6. RESET 2 MIN
1427 C206       CLR SIL
                CHK_HOLD:  ;;STOP NEB?
1429 300809     JNB HOLD,SAMPLE
142C DF07       DJNZ RHOLD_TIM,SAMPLE ;R7
142E 7F78       MOV RHOLD_TIM,#120
1430 200602     JB SIL,SAMPLE
1433 D2A7       SETB P2.7    ;ON BUZZER
```

```
                    SAMPLE:    ::READ VENT SIG
 1435 C2D3          CLR PSW.3    :BANK 0
 1437 C2D4          CLR PSW.4
 1439 53A0F8        ANL P2.#11111000B    :CLEAR MUX ADDRESS
 143C D2A3          SETB P2.3
 143E               ANALOG RVENT_SIG
+143E 00            NOP  :DELAY TIME FOR MUX
+143F 00            NOP
+1440 00            NOP
+1441 00            NOP
+1442 00            NOP
+1443 C2A3          CLR P2.3      :START CONVERSION
+1445 00            NOP      :ALLOW CONV. TIME 5 MICROSEC
+1446 00            NOP
+1447 00            NOP
+1448 A990          MOV RVENT_SIG,P1      :SAVE DIGITAL OUTPUT
+144A D2A3          SETB P2.3
 144C 00            NOP
 144D               RUNNING_AVG VENT_LOW,4,RVENT_SIG,VENT_AVG
+                   :CALCULATES RUNNING AVERAGE OF 4 BYTES IN DATA MEMORY
+                   :WITH A LOW ADDRESS OF LODATA. INPUT SIGNAL IS LOCATED
+                   :AT INSIG. AVERAGE OUTPUT IS AT AVG.
+144D C0D0          PUSH PSW
+144F C0E0          PUSH ACC
+1451 C0F0          PUSH B
+1453 C2D3          CLR PSW.3   :BANK0
+1455 C2D4          CLR PSW.4
+1457 7468          MOV A,#VENT_LOW    :SET R0
+1459 2404          ADD A,#4
+145B 14            DEC A
+145C F8            MOV R0,A
+                   NEXT10004:
+145D 18            DEC R0
+145E E6            MOV A,@R0     :SHIFT UP
+145F 08            INC R0
+1460 F6            MOV @R0,A
+1461 18            DEC R0
+1462 8868F8        CJNE R0,#VENT_LOW,NEXT10004   :LODATA ADDRESS
+1465 E9            MOV A,RVENT_SIG     :MOV NEW DATA TO VENT_LOW
+1466 75F004        MOV B,#4
+1469 84            DIV AB
+146A F6            MOV @R0,A
+146B 7468          MOV A,#VENT_LOW   :ADD TO CALC VENT_AVG
+146D 2404          ADD A,#4
+146F 14            DEC A
+1470 F51C          MOV TEMP_STORE,A
+1472 E6            MOV A,@R0
+1473 C8            XCH A,R0
+                   NEXT20004:
+1474 C8            XCH A,R0
+1475 08            INC R0
+1476 26            ADD A,@R0
+1477 C8            XCH A,R0
+1478 951CF9        CJNE A,TEMP_STORE,NEXT20004
```

```
+1478 C8          XCH A,R0
+147C F56C        MOV VENT_AVG,A
+147E D0F0        POP B
+1480 D0E0        POP ACC
+1482 D0D0        POP PSW
 1484 00          NOP
 1485 05A0        INC P2
 1487             ANALOG RFLT_FLO
+1487 00          NOP  :DELAY TIME FOR MUX
+1488 00          NOP
+1489 00          NOP
+148A 00          NOP
+148B 00          NOP
+148C C2A3        CLR P2.3      :START CONVERSION
+148E 00          NOP      :ALLOW CONV. TIME 5 MICROSEC
+148F 00          NOP
+1490 00          NOP
+1491 AA90        MOV RFLT_FLO,P1      :SAVE DIGITAL OUTPUT
+1493 D2A3        SETB P2.3
 1495 00          NOP
 1496             RUNNING_AVG FLTFLO_LO,4,RFLT_FLO,FLTFLO_AVG
+                 :CALCULATES RUNNING AVERAGE OF 4 BYTES IN DATA MEMORY
+                 :WITH A LOW ADDRESS OF LODATA. INPUT SIGNAL IS LOCATED
+                 :AT INSIG. AVERAGE OUTPUT IS AT AVG.
+1496 C0D0        PUSH PSW
+1498 C0E0        PUSH ACC
+149A C0F0        PUSH B
+149C C2D3        CLR PSW.3  :BANK0
+149E C2D4        CLR PSW.4
+14A0 7440        MOV A,#FLTFLO_LO    :SET R0
+14A2 2404        ADD A,#4
+14A4 14          DEC A
+14A5 F8          MOV R0,A
+                 NEXT10006:
+14A6 18          DEC R0
+14A7 E6          MOV A,@R0     :SHIFT UP
+14A8 08          INC R0
+14A9 F6          MOV @R0,A
+14AA 18          DEC R0
+14AB B840F8      CJNE R0,#FLTFLO_LO,NEXT10006  :LODATA ADDRESS
+14AE EA          MOV A,RFLT_FLO     :MOV NEW DATA TO FLTFLO_LO
+14AF 75F004      MOV B,#4
+1482 84          DIV AB
+1483 F6          MOV @R0,A
+1484 7440        MOV A,#FLTFLO_LO  :ADD TO CALC FLTFLO_AVG
+1486 2404        ADD A,#4
+1488 14          DEC A
+1489 F51C        MOV TEMP_STORE,A
+148B E6          MOV A,@R0
+148C C8          XCH A,R0
+                 NEXT20006:
+148D C8          XCH A,R0
+148E 08          INC R0
+148F 26          ADD A,@R0
+14C0 C8          XCH A,R0
```

```
+14C1 B51CF9    CJNE A,TEMP_STORE,NEXT20006
+14C4 C8        XCH A,R0
+14C5 F544      MOV FLTFLO_AVG,A
+14C7 D0F0      POP B
+14C9 D0E0      POP ACC
+14CB D0D0      POP PSW
 14CD 00        NOP
 14CE 05A0      INC P2
 14D0           ANALOG RAW_PRESS
+14D0 00        NOP  :DELAY TIME FOR MUX
+14D1 00        NOP
+14D2 00        NOP
+14D3 00        NOP
+14D4 00        NOP
+14D5 C2A3      CLR P2.3      :START CONVERSION
+14D7 00        NOP      :ALLOW CONV. TIME 5 MICROSEC
+14D8 00        NOP
+14D9 00        NOP
+14DA A890      MOV RAW_PRESS,P1      :SAVE DIGITAL OUTPUT
+14DC D2A3      SETB P2.3
 14DE 00        NOP
 14DF           RUNNING_AVG AWP_LO,4,RAW_PRESS,AWP_AVG
+              :CALCULATES RUNNING AVERAGE OF 4 BYTES IN DATA MEMORY
+              :WITH A LOW ADDRESS OF LODATA. INPUT SIGNAL IS LOCATED
+              :AT INSIG. AVERAGE OUTPUT IS AT AVG.
+14DF C0D0      PUSH PSW
+14E1 C0E0      PUSH ACC
+14E3 C0F0      PUSH B
+14E5 C2D3      CLR PSW.3   :BANK0
+14E7 C2D4      CLR PSW.4
+14E9 7448      MOV A,#AWP_LO    :SET R0
+14EB 2404      ADD A,#4
+14ED 14        DEC A
+14EE F8        MOV R0,A
+              NEXT10008:
+14EF 18        DEC R0
+14F0 E6        MOV A,@R0     :SHIFT UP
+14F1 08        INC R0
+14F2 F6        MOV @R0,A
+14F3 18        DEC R0
+14F4 B848F8    CJNE R0,#AWP_LO,NEXT10008   :LODATA ADDRESS
+14F7 EB        MOV A,RAW_PRESS     :MOV NEW DATA TO AWP_LO
+14F9 75F004    MOV B,#4
+14FB 84        DIV AB
+14FC F6        MOV @R0,A
+14FD 7448      MOV A,#AWP_LO   :ADD TO CALC AWP_AVG
+14FF 2404      ADD A,#4
+1501 14        DEC A
+1502 F51C      MOV TEMP_STORE,A
+1504 E6        MOV A,@R0
+1505 C8        XCH A,R0
+              NEXT20008:
+1506 C8        XCH A,R0
+1507 08        INC R0
+1508 26        ADD A,@R0
```

```
+1509 C8          XCH A,R0
+150A B51CF9      CJNE A,TEMP_STORE,NEXT20008
+150D C8          XCH A,R0
+150E F54C        MOV AWP_AVG,A
+1510 D0F0        POP B
+1512 D0E0        POP ACC
+1514 D0D0        POP PSW
 1516 00          NOP
 1517 301509      JNB INSP,NEXT_SAMP
 151A E54D        MOV A,AWP_MAX
 151C C3          CLR C
 151D 954C        SUBB A,AWP_AVG
 151F 5003        JNC NEXT_SAMP
 1521 854C4D      MOV AWP_MAX,AWP_AVG
                  NEXT_SAMP:
 1524 00          NOP
 1525 05A0        INC P2
 1527             ANALOG RNEB_FLO
+1527 00          NOP  :DELAY TIME FOR MUX
+1528 00          NOP
+1529 00          NOP
+152A 00          NOP
+152B 00          NOP
+152C C2A3        CLR P2.3      :START CONVERSION
+152E 00          NOP      ;ALLOW CONV. TIME 5 MICROSEC
+152F 00          NOP
+1530 00          NOP
+1531 AC90        MOV RNEB_FLO,P1      ;SAVE DIGITAL OUTPUT
+1533 D2A3        SETB P2.3
 1535 00          NOP
 1536 EC          MOV A,RNEB_FLO  :R4
 1537 C3          CLR C
 1538 9432        SUBB A,#50
 153A 400E        JC NEGFLO
 153C C3          CLR C  :DIV BY 4
 153D 13          RRC A
 153E C3          CLR C
 153F 13          RRC A
 1540 2550        ADD A,POSSUM :SUN POS FLOW
 1542 F550        MOV POSSUM,A  :SAVE
 1544 500F        JNC CONT1
 1546 D20F        SETB FLOW  :OVERFLOW CONDITION
 1548 8008        SJMP CONT1
                  NEGFLO:  ::NEG FLOW
 154A 7432        MOV A,#50
 154C 9C          SUBB A,RNEB_FLO
 154D C3          CLR C  :DIV BY 4
 154E 13          RRC A
 154F C3          CLR C
 1550 13          RRC A
 1551 2551        ADD A,NEGSUM
 1553 F551        MOV NEGSUM,A  :SAVE
                  CONT1:
 1555 00          NOP
 1556 05A0        INC P2
```

```
 1558                ANALOG RTEMP
+1558 00            NOP   :DELAY TIME FOR MUX
+1559 00            NOP
+155A 00            NOP
+155B 00            NOP
+155C 00            NOP
+155D C2A3          CLR P2.3       :START CONVERSION
+155F 00            NOP      :ALLOW CONV. TIME 5 MICROSEC
+1560 00            NOP
+1561 00            NOP
+1562 AD90          MOV RTEMP.P1       :SAVE DIGITAL OUTPUT
+1564 D2A3          SETB P2.3
 1566 00            NOP
                    RETURN:   ::RET FROM INT
 1567 D0D0          POP PSW
 1569 D0F0          POP B
 156B D0E0          POP ACC
 156D 32            RETI


                    SERVICE:  ::CHK FLOW,SER-REC,BLINK
 156E 103902        JBC CLK.TEMP_CONT
 1571 22            RET
 1572 00            NOP
                    TEMP_CONT:   ::CONTROL HEATER
 1573 C2D3          CLR PSW.3   :BANK0
 1575 C2D4          CLR PSW.4
 1577 ED            MOV A,RTEMP   :RS
 1578 B4A00C        CJNE A,#TEMP_HI,NOT_EQ
                    HITEMP:   ::OVER 80C
 157B C281          CLR P0.1   :HEAT OFF
 157D C23A          CLR HEAT
 157F D220          SETB L35 :HI TEMP LED
 1581 852599        MOV SBUF,LED3
 1584 D20C          SETB ALM
 1586 22            RET
 1587 5022          NOT_EQ: JNC HI_TEMP :RTEMP>TEMP_HI
 1589 E511          MOV A,TEMP_SET
 158B B42804        CJNE A,#40,HEAT_CHK
 158E C281          CLR P0.1   :HEAT OFF
 1590 A185          AJMP FLO_TST
                    HEAT_CHK:  ::CHK HEAT BIT
 1592 203A0C        JB HEAT,SW_OFF
 1595 C3            CLR C
 1596 940A          SUBB A,#10   :LOW LIMIT
 1598 9D            SUBB A,RTEMP :RS
 1599 401A          JC FLO_TST   :?LEAVE OFF?
 159B D281          SETB P0.1   :TURN ON
 159D D23A          SETB HEAT
 159F A185          AJMP FLO_TST
                    SW_OFF:
 15A1 240A          ADD A,#10   :UPPER LIMIT
 15A3 C3            CLR C
 15A4 9D            SUBB A,RTEMP
 15A5 500E          JNC FLO_TST   :?LEAVE ON?
```

```
15A7 C281       CLR P0.1   :TURN OFF
15A9 C23A       CLR HEAT
                HI_TEMP:   ::TEMP ALARM
15AB D220       SETB L35   :HI TEMP LED
15AD 852599     MOV SBUF,LED3
15B0 D125       ACALL TRANS_DEL
15B2 D20C       SETB ALM
15B4 22         RET
                FLO_TST:   ::TEST NEB FLOW
15B5 200070     JB WAIT.CHK_SERPORT
15B8 D2D3       SETB PSW.3   ;BANK1
15BA C2D4       CLR PSW.4
15BC 100F1C     JBC FLOW.FLO
15BF E550       MOV A,POSSUM
15C1 C3         CLR C
15C2 9551       SUBB A,NEGSUM :CALC SFLO
15C4 5004       JNC CONT2
15C6 DD17       DJNZ ROFF_TIM,CONT4
15C8 A1F7       AJMP NOFLO_ALM
                CONT2:
15CA F5F0       MOV B,A   :SAVE SFLO=POS-NEG
15CC 948C       SUBB A,#NOFLO_TH   :SFLO-THRESH
15CE 5004       JNC CONT3
15D0 DD02       NOFLO:  DJNZ ROFF_TIM,CONT3
15D2 A1F7       AJMP NOFLO_ALM
                CONT3:
15D4 E5F0       MOV A,B   :SFLO
15D6 C3         CLR C
15D7 942D       SUBB A,#FLO_TH   :SFLO-THRESH
15D9 4004       JC CONT4   ;?SFLO<THRESH
15DB DC02       FLO: DJNZ RON_TIM,CONT4
15DD C10E       AJMP FLO_ALM
                CONT4:   ::CHECK TIME
15DF 755000     MOV POSSUM,#0   :RESET FLOW SUM
15E2 755100     MOV NEGSUM,#0
15E5 D51405     DJNZ ONTIMER,CHK_OFFTIM
15E8 751408     MOV ONTIMER,#FLOTIM
15EB 7C08       MOV RON_TIM,#FLOTIM
                CHK_OFFTIM:
15ED D51538     DJNZ OFFTIMER,CHK_SERPORT
15F0 751532     MOV OFFTIMER,#NOFLOTIM
15F3 7D32       MOV ROFF_TIM,#NOFLOTIM
15F5 C128       AJMP CHK_SERPORT

                NOFLO_ALM:  :?NEB OFF > 10S
15F7 755000     MOV POSSUM,#0
15FA 755100     MOV NEGSUM,#0
15FD 751532     MOV OFFTIMER,#NOFLOTIM
1600 7D32       MOV ROFF_TIM,#NOFLOTIM
1602 D214       SETB BEEP
1604 D20C       SETB ALM
1606 D234       SETB L24    :NO FLOW LED
1608 852699     MOV SBUF,LED2
160B D125       ACALL TRANS_DEL
160D 22         RET
```

```
                    FLO_ALM:    ::NEB ON > 2.2S
160E 755000         MOV POSSUM,#0
1611 755100         MOV NEGSUM,#0
1614 751408         MOV ONTIMER,#FLOTIM
1617 7508           MOV RON_TIM,#FLOTIM
1619 D20C           SETB ALM    :FLAG
161B D237           SETB L27   :CONT FLOW ALM
161D 8526A9         MOV SBUF,LED2
1620 D125           ACALL TRANS_DEL
1622 22             RET

1623 613F           BLINK_BEEP1:   AJMP BLINK_BEEP
                    TRANS_DEL:  ::DELAY 2.25MS,CC=80EH
1625 02198E         LJMP TRANS_DEL1

                    CHK_SERPORT:   ::NEW CHAR REC?
1628 3098F8         JNB RI,BLINK_BEEP1
162B C298           CLR RI
162D C2A9           CLR ET0   ;DISABLE TIMER 0 INT
162F C28C           CLR TR0   :DISABLE TIMER 0
1631 E599           MOV A,SBUF    :READ CODE RECEIVED
1633 C4             SWAP A
1634 23             RL A    :MULTIPLY BY 2
1635 901639         MOV DPTR,#JUMP_TBLE1
1638 73             JMP @A+DPTR
1639 C17D           JUMP_TBLE1: AJMP CASE0    ;TEMP. SET
163B C1F4                       AJMP CASE1    :NEB. HOLD
163D E19F                       AJMP CASE2    :SELF TEST
163F E19D                       AJMP CASE3    :NO ACTION
1641 E126                       AJMP CASE4    :VENT SEL
1643 E1AE                       AJMP CASE5    :DISPLAY TEMP
1645 C15D                       AJMP CASE61   :ALM SIL
1647 C161                       AJMP CASE71   :NO ACTION
1649 E169                       AJMP CASE8    :CHANGE VOL
164B C165                       AJMP CASE91   ;DISPLAY FILT LD
164D C169                       AJMP CASEA1   :ALM RESET
164F C16D                       AJMP CASEB1   :NO ACTION
1651 C171                       AJMP CASEC1   :ENTER
1653 C175                       AJMP CASED1   :DISPLAY PIP
1655 C179                       AJMP CASEE1   :ALM TEST
1657 C15A                       AJMP CASEF    :NO ACTION
1659 00             NOP
                    CASEF:  :NO ACTION
165A 613F           AJMP BLINK_BEEP
165C 00             NOP
165D 021860         CASE61: LJMP CASE6
1660 00             NOP
1661 021867         CASE71: LJMP CASE7
1664 00             NOP
1665 021818         CASE91: LJMP CASE9
1668 00             NOP
1669 02187A         CASEA1: LJMP CASEA
166C 00             NOP
166D 02186A         CASEB1: LJMP CASEB
```

```
1670 00          NOP
1671 021843      CASEC1: LJMP CASEC
1674 00          NOP
1675 0218D7      CASED1: LJMP CASED
1678 00          NOP
1679 021955      CASEE1: LJMP CASEE
167C 00          NOP

                 CASE0:  ::TEMP SET
167D 203819      JB TEMP.NEW_TEMP
1680 D238        SETB TEMP
1682 E511        MOV A.TEMP_SET
1684 B42820      CJNE A,#40.DISPLAY_TEMP
                 OFF_STATE: ::LCD "- -"
1687 7599F0      MOV SBUF.#0F0H :HUNS BLANK
168A D125        ACALL TRANS_DEL
168C 7599A1      MOV SBUF.#0A1H :TENS "-"
168F D125        ACALL TRANS_DEL
1691 7599A2      MOV SBUF.#0A2H :ONES "-"
1694 D125        ACALL TRANS_DEL
1696 613F        AJMP BLINK_BEEP
1698 00          NOP
                 NEW_TEMP: ::NEXT SET TEMP
1699 E511        MOV A.TEMP_SET
1698 B47805      CJNE A.#120.CALC_TEMP
169E 751128      MOV TEMP_SET.#40
16A1 C187        AJMP OFF_STATE

                 CALC_TEMP:
16A3 2414        ADD A,#20
16A5 F511        MOV TEMP_SET.A

                 DISPLAY_TEMP:
16A7 C3          CLR C
16A8 13          RRC A  :DIV BY 2
16A9             BINARY_BCD DEC_HUN.DEC_TEN.DEC_ONE
+                :CONVERTS BYTE LOCATED IN ACC TO  DECIMAL
+                :AND STORES RESULT IN DEC_HUN. DEC_TEN AND ONE.
+16A9 752900      MOV DEC_HUN.#0  :CLEAR REGISTERS
+16AC 752A00      MOV DEC_TEN.#0
+16AF 752B00      MOV DEC_ONE.#0
+                 CALC_HUN0011:  ::SUBTRACT 100
+16B2 F5F0        MOV B.A
+                 NEXTSUB100011:
+16B4 C3          CLR C
+16B5 9464        SUBB A.#100
+16B7 4006        JC CALC_TEN0011
+16B9 0529        INC DEC_HUN
+16BB F5F0        MOV B.A     ;SAVE
+16BD 80F5        SJMP NEXTSUB100011
+                 CALC_TEN0011:  ::SUBTRACT 10
+16BF E5F0        MOV A.B
+                 NEXTSUB20011:
+16C1 C3          CLR C
+16C2 940A        SUBB A.#10
```

```
+16C4 4006      JC CALC_ONE0011
+16C6 052A      INC DEC_TEN
+16C8 F5F0      MOV B,A
+16CA 80F5      SJMP NEXTSUB20011
+               CALC_ONE0011:
+16CC 85F028    MOV DEC_ONE,B
+16CF E529      MOV A,DEC_HUN
+16D1 700A      JNZ BCD_OUT0011
+16D3 752F0F    MOV DEC_HUN,#0FH   ;BLANK
+16D6 E52A      MOV A,DEC_TEN
+16D8 7003      JNZ BCD_OUT0011
+16DA 752A0F    MOV DEC_TEN,#0FH   ;BLANK
+               BCD_OUT0011:
 16DD 7599F0    MOV SBUF,#0F0H   ;HUN BLANK
 16E0 D125      ACALL TRANS_DEL
 16E2 E52A      MOV A,DEC_TEN
 16E4 C4        SWAP A
 16E5 4401      ORL A,#01H
 16E7 F512      MOV TEMP_DEC,A   ;SAVE TENS
 16E9 F599      MOV SBUF,A
 16EB D125      ACALL TRANS_DEL
 16ED 759902    MOV SBUF,#02H   ;ONES
 16F0 D125      ACALL TRANS_DEL
 16F2 613F      AJMP BLINK_BEEP

                CASE1:    ;:NEBULIZER HOLD
 16F4 D2D3      SETB PSW.3    ;BANK1
 16F6 C2D4      CLR PSW.4
 16F8 100810    JBC HOLD,HOLD_OFF
 16FB D208      SETB HOLD    ;HOLD FLAG
 16FD D214      SETB BEEP
 16FF D235      SETB L25     ;NEB HOLD LED
 1701 852699    MOV SBUF,LED2
 1704 D125      ACALL TRANS_DEL
 1706 43A070    ORL P2,#01110000B    ;OFF VALVES
 1709 8008      SJMP HOLD_OUT
                HOLD_OFF:
 170B C208      CLR HOLD    ;HOLD FLAG
 170D C214      CLR BEEP
 170F C235      CLR L25     ;OFF HOLD LED
 1711 852699    MOV SBUF,LED2
 1714 D125      ACALL TRANS_DEL
                HOLD_OUT:
 1716 7F78      MOV RHOLD_TIM,#120 ;R7 RESET
 1718 D200      SETB WAIT
 171A D21E      SETB L16 ;WAIT LED
 171C 852399    MOV SBUF,LED1
 171F D125      ACALL TRANS_DEL
 1721 752F00    MOV THREE_CYCLE,#0
 1724 613F      AJMP BLINK_BEEP

                CASE4:    ::SELECT VENT
 1726 43A070    ORL P2,#01110000B   ;VALVES OFF
 1729 D204      SETB VEN_SEL
 1728 D200      SETB WAIT
```

```
172D 752F00     MOV THREE_CYCLE,#0
1730 D21E       SETB L16  ;WAIT
1732 852399     MOV SBUF,LED1
1735 D125       ACALL TRANS_DEL
1737 C2D3       CLR PSW.3    ;BANK0
1739 C2D4       CLR PSW.4
173B EE         MOV A,RVENT    ;R4, INC. VENT. NO.
173C 2410       ADD A,#10H
173E B44302     CJNE A,#43H,SEE_VENT
1741 7413       MOV A,#13H  ;RESET #1
                SEE_VENT:
1743 FE         MOV RVENT,A
1744 8E99       MOV SBUF,RVENT    ;DISPLAY NEW NUMBER
1746 D125       ACALL TRANS_DEL
1748 00         NOP
1749 EE         MOV A,RVENT    ;LOOK UP THRESHOLDS FOR VENTILATOR SELECTED
174A C4         SWAP A
174B 540F       ANL A,#0FH    ;CLEAR ADDRESS
174D 23         RL A    ;MULT. BY 2
174E F5F0       MOV B,A     ;SAVE
1750 F15D       ACALL VENT_TBLE
1752 F518       MOV VENT_HI,A    ;STORE UPPER THRESH
1754 E5F0       MOV A,B
1756 14         DEC A
1757 F15D       ACALL VENT_TBLE
1759 F51A       MOV VENT_LO,A    ;STORE LOWER THRESH
175B 613F       AJMP BLINK_BEEP

175D 83         VENT_TBLE: MOVC A,@A+PC
175E 22         RET    ;THRESHOLDS
175F 38 45 81   DB 38H,45H,81H,86H,38H,45H ;SERVO LO 2.3V, HI 2.7V
1762 86 38 45
                ;P87200 LO 5.05V, HI 5.25V,  HAM LO 2.3V, HI 2.7V

                CASE81: ::INITIALIZATION ENTRY
1765 C2A9       CLR ET0
1767 C28C       CLR TR0

                CASE8:    ;::CHANGE VOLUME
1769 D203       SETB VOL_CHG
176B E528       MOV A,CHG_VOL
176D C4         SWAP A
176E 23         RL A
176F F5F0       MOV B,A
1771 F18F       ACALL CHGVOL_TBLE
1773 F519       MOV SET_CHGTIM,A
1775 E5F0       MOV A,B
1777 14         DEC A
1778 F18F       ACALL CHGVOL_TBLE
177A F529       MOV CHG_VOL,A
177C 852399     MOV SBUF,CHG_VOL
177F D125       ACALL TRANS_DEL
1781 D200       SETB WAIT
1783 752F00     MOV THREE_CYCLE,#0
1786 D21E       SETB L16
```

```
1788 852399     MOV SBUF,LED1
178B D125       ACALL TRANS_DEL
178D 613F       AJMP BLINK_BEEP
                CHGVOL_TBLE:   ::SELECT NEW VOL
178F 83         MOVC A,@A+PC
1790 22         RET
1791 20 14 40   DB 20H,20,40H,40,0,0,60H,60,0,0,10H,10
1794 28 00 00 60 3C 00 00 10 0A
                    :SHIFT TO NEW VOLUME

                CASE3:    ::NO ACTION
179D 613F       AJMP BLINK_BEEP


                CASE2: :NO ACTION
179F 613F       AJMP BLINK_BEEP

                CASE5:   ::DISPLAY TEMP
17A1 101058     JBC SEE_TEMP,RESTORE_VOL1
17A4 D210       SETB SEE_TEMP
17A6 C2D3       CLR PSW.3   :BANK0
17A8 C2D4       CLR PSW.4
17AA ED         MOV A,RTEMP :R5
17AB C3         CLR C
17AC 13         RRC A   ;DIV BY 2
17AD            BINARY_BCD DEC_HUN,DEC_TEN,DEC_ONE
+               ;CONVERTS BYTE LOCATED IN ACC TO  DECIMAL
+               :AND STORES RESULT IN DEC_HUN, DEC_TEN AND ONE.
+17AD 752900     MOV DEC_HUN,#0   :CLEAR REGISTERS
+17B0 752A00     MOV DEC_TEN,#0
+17B3 752B00     MOV DEC_ONE,#0
+               CALC_HUN0012:   ::SUBTRACT 100
+17B6 F5F0       MOV B,A
+               NEXTSUB10012:
+17B8 C3         CLR C
+17B9 9464       SUBB A,#100
+17BB 4006       JC CALC_TEN0012
+17BD 0529       INC DEC_HUN
+17BF F5F0       MOV B,A     :SAVE
+17C1 80F5       SJMP NEXTSUB10012
+               CALC_TEN0012:   ::SUBTRACT 10
+17C3 E5F0       MOV A,B
+               NEXTSUB20012:
+17C5 C3         CLR C
+17C6 940A       SUBB A,#10
+17C8 4006       JC CALC_ONE0012
+17CA 052A       INC DEC_TEN
+17CC F5F0       MOV B,A
+17CE 80F5       SJMP NEXTSUB20012
+               CALC_ONE0012:
+17D0 85F02B     MOV DEC_ONE,B
+17D3 E529       MOV A,DEC_HUN
+17D5 700A       JNZ BCD_OUT0012
+17D7 75290F     MOV DEC_HUN,#0FH   :BLANK
+17DA E52A       MOV A,DEC_TEN
```

```
+17DC 7003      JNZ BCD_OUTOO12
+17DE 752A0F    MOV DEC_TEN.#0FH   :BLANK
+               BCD_OUTOO12:
 17E1 00        NOP
 17E2 E529      MOV A.DEC_HUN
 17E4 C4        SWAP A
 17E5 F599      MOV SBUF.A
 17E7 D125      ACALL TRANS_DEL
 17E9 E52A      MOV A.DEC_TEN
 17EB C4        SWAP A
 17EC 4401      ORL A.#01H
 17EE F599      MOV SBUF.A
 17F0 D125      ACALL TRANS_DEL
 17F2 E528      MOV A.DEC_ONE
 17F4 C4        SWAP A
 17F5 4402      ORL A.#02H
 17F7 F599      MOV SBUF.A
 17F9 121625    LCALL TRANS_DEL
 17FC 02133F    LJMP BLINK_BEEP

                RESTORE_VOL1:  ::DISPLAY VOL
 17FF C2D3      CLR PSW.3   :BANK2
 1801 D2D4      SETB PSW.4
 1803 852899    MOV SBUF.CHG_VOL
 1806 121625    LCALL TRANS_DEL
 1809 759901    MOV SBUF,#01H
 180C 121625    LCALL TRANS_DEL
 180F 759902    MOV SBUF,#02H
 1812 121625    LCALL TRANS_DEL
                OUT_TEMP:
 1815 02133F    LJMP BLINK_BEEP

                CASE9:  ::DISPLAY FLT LOAD
 1818 101113    JBC SEE_LD.RESTORE_VOL2
 1818 D211      SETB SEE_LD
 181D 852C99    MOV SBUF.FLTLD_HUN
 1820 121625    LCALL TRANS_DEL
 1823 852D99    MOV SBUF.FLTLD_TEN
 1826 121625    LCALL TRANS_DEL
 1829 852E99    MOV SBUF.FLTLD_ONE
 182C 0140      AJMP OUT_DISPLD
                RESTORE_VOL2:  ::DISPLAY VOL
 182E 852899    MOV SBUF.CHG_VOL
 1831 121625    LCALL TRANS_DEL
 1834 759901    MOV SBUF.#01H
 1837 121625    LCALL TRANS_DEL
 183A 759902    MOV SBUF.#02H
 183D 121625    LCALL TRANS_DEL
                OUT_DISPLD:
 1840 02133F    LJMP BLINK_BEEP

                CASEC:   ::ENTER KEY
 1843 C204      CLR VEN_SEL
 1845 C203      CLR VOL_CHG
 1847 C23R      CLR TEMP
```

```
1849 C2D3       CLR PSW.3   :BANKO
184B C2D4       CLR PSW.4
184D 8E99       MOV SBUF,RVENT
184F 121625     LCALL TRANS_DEL
1852 852899     MOV SBUF,CHG_VOL   :SET HUNS
1855 121625     LCALL TRANS_DEL
1858 759901     MOV SBUF,#01H   :SET TENS
185B 121625     LCALL TRANS_DEL
185E 759902     MOV SBUF,#02H   :SET ONES
1861 121625     LCALL TRANS_DEL
1864 02133F     LJMP BLINK_BEEP

                CASE7:    ::NO ACTION
1867 02133F     LJMP BLINK_BEEP

                CASE8:    ::NO ACTION
186A 02133F     LJMP BLINK_BEEP

                CASE6:    ::SIL ALM 2 MIN
186D D2D3       SETB PSW.3   ;BANK1
186F C2D4       CLR PSW.4
1871 C2A7       CLR P2.7    ;OFF BUZZER
1873 D206       SETB SIL    :SILENCE FLAG
1875 7E78       MOV RSIL_TIM,#120    :R6,TWO MIN. TIMER
1877 02133F     LJMP BLINK_BEEP

                CASEA:    ::ALM RST
187A 75A078     MOV P2,#78H   :OUTPUTS OFF
187D 752000     MOV 20H,#0   ;CLEAR BITS
1880 752100     MOV 21H,#0
1883 752200     MOV 22H,#0
1886 752700     MOV 27H,#0
1889 D200       SETB WAIT
188B D21E       SETB L16   :WAIT
188D C21D       CLR L15   ;FILT CHANGE
188F C21F       CLR L17   :LOFLOW
1891 852399     MOV SBUF,LED1
1894 318E       ACALL TRANS_DEL1
1896 852899     MOV SBUF,CHG_VOL  :NORMAL LCD
1899 318E       ACALL TRANS_DEL1
189B 759901     MOV SBUF,#01H
189E 318E       ACALL TRANS_DEL1
18A0 759902     MOV SBUF,#02H
18A3 318E       ACALL TRANS_DEL1
18A5 C2D3       CLR PSW.3   ;BANKO
18A7 C2D4       CLR PSW.4
18A9 8E99       MOV SBUF,RVENT  :R1
18AB 318E       ACALL TRANS_DEL1
18AD 53260F     ANL LED2,#0FH  :OFF
18B0 852699     MOV SBUF,LED2
18B3 318E       ACALL TRANS_DEL1
18B5 C22C       CLR L34   :HI PRESS
18B7 852599     MOV SBUF,LED3
18BA 318E       ACALL TRANS_DEL1
18BC 752F00     MOV THREE_CYCLE,#0   :RESET
```

```
188F D2D3       SETB PSW.3   :BANK1
18C1 C2D4       CLR PSW.4
18C3 7C00       MOV RON_TIM.#0
18C5 7D00       MOV ROFF_TIM.#0
18C7 755000     MOV POSSUM.#0
18CA 755100     MOV NEGSUM.#0
18CD 758C70     MOV TH0.#70H   :RST TIMER
18D0 75A887     MOV IE.#87H    :SET ET0
18D3 758850     MOV TCON.#50H  :SET TR0
18D6 22         RET

                CASED:    ::DISPLAY PIP
                  :MULTIPLY BY SCALE FACTOR OF 5/8. CONVERT TO BCD
                  :AND DISPLAY PIP. RETURN TO VOLUME DISPLAY WHEN
                  :SWITCH IS PRESSED A SECOND TIME.

18D7 100968     JBC SEE_PIP.LCD_VOL
18DA D209       SETB SEE_PIP
18DC E54E       MOV A.PIP_STORE
18DE 75F005     MOV B.#5
18E1 A4         MUL AB     :MSB IN B
18E2 C5F0       XCH A.B    :RRC 3 TIMES TO DIVIDE BY 8
18E4 13         RRC A       :MSB IN A
18E5 C5F0       XCH A.B    :LSB IN A
18E7 13         RRC A
18E8 C3         CLR C      ;SECOND ROTATION
18E9 C5F0       XCH A.B
18EB 13         RRC A
18EC C5F0       XCH A.B
18EE 13         RRC A
18EF C3         CLR C      :THIRD ROTATION
18F0 C5F0       XCH A.B
18F2 13         RRC A
18F3 C5F0       XCH A.B
18F5 13         RRC A
18F6 9414       SUBB A.#14H   :ZERO OFFSET
18F8            BINARY_BCD DEC_HUN,DEC_TEN,DEC_ONE
+               :CONVERTS BYTE LOCATED IN ACC TO  DECIMAL
+               ;AND STORES RESULT IN DEC_HUN, DEC_TEN AND ONE.
+18F8 752900     MOV DEC_HUN.#0   :CLEAR REGISTERS
+18FB 752A00     MOV DEC_TEN.#0
+18FE 752B00     MOV DEC_ONE.#0
+               CALC_HUN0013:   ::SUBTRACT 100
+1901 F5F0       MOV B.A
+               NEXTSUB10013:
+1903 C3         CLR C
+1904 9464       SUBB A.#100
+1906 4006       JC CALC_TEN0013
+1908 0529       INC DEC_HUN
+190A F5F0       MOV B.A     :SAVE
+190C 80F5       SJMP NEXTSUB10013
+               CALC_TEN0013:   ::SUBTRACT 10
+190E E5F0       MOV A.B
+               NEXTSUB20013:
+1910 C3         CLR C
```

```
+1911 940A      SUBB A,#10
+1913 4006      JC CALC_ONE0013
+1915 052A      INC DEC_TEN
+1917 F5F0      MOV B,A
+1919 80F5      SJMP NEXTSUB20013
+               CALC_ONE0013:
+191B 85F02B    MOV DEC_ONE,B
+191E E529      MOV A,DEC_HUN
+1920 700A      JNZ BCD_OUT0013
+1922 75290F    MOV DEC_HUN,#0FH   ;BLANK
+1925 E52A      MOV A,DEC_TEN
+1927 7003      JNZ BCD_OUT0013
+1929 752A0F    MOV DEC_TEN,#0FH   ;BLANK
+               BCD_OUT0013:
 192C E529      MOV A,DEC_HUN ;DISPLAY PIP
 192E C4        SWAP A
 192F F599      MOV SBUF,A
 1931 318E      ACALL TRANS_DEL1
 1933 E52A      MOV A,DEC_TEN
 1935 C4        SWAP A
 1936 4401      ORL A,#01H
 1938 F599      MOV SBUF,A
 193A 318E      ACALL TRANS_DEL1
 193C E528      MOV A,DEC_ONE
 193E C4        SWAP A
 193F 4402      ORL A,#02H
 1941 F599      MOV SBUF,A
 1943 800D      SJMP OUTPIP
                LCD_VOL:   ::DISPLAY VOL
 1945 852899    MOV SBUF,CHG_VOL
 1948 318E      ACALL TRANS_DEL1
 194A 759901    MOV SBUF,#01H
 194D 318E      ACALL TRANS_DEL1
 194F 759902    MOV SBUF,#02H
                OUTPIP:
 1952 02133F    LJMP BLINK_BEEP

                CASEE:    ::ALM TEST
                :PUSH SW TO TEST & PUSH TO RETURN
 1955 100A33    JBC ALM_TST,NORMAL
 1958 D20A      SETB ALM_TST
 195A D2A7      SETB P2.7    ;ON BUZZER
 195C 7480      MOV A,#80H
 195E F599      MOV SBUF,A  ;HUNS LCD TEST
 1960 12198E    LCALL TRANS_DEL1
 1963 04        INC A        ;TENS
 1964 F599      MOV SBUF,A
 1966 12198E    LCALL TRANS_DEL1
 1969 04        INC A        ;ONES
 196A F599      MOV SBUF,A
 196C 12198E    LCALL TRANS_DEL1
 196F 04        INC A        ;VENT #
 1970 F599      MOV SBUF,A
 1972 12198E    LCALL TRANS_DEL1
 1975 74F4      MOV A,#0F4H ;LED1 TEST
```

```
1977 F599       MOV SBUF,A
1979 12198E     LCALL TRANS_DEL1
197C 04         INC A          ;LED2
197D F599       MOV SBUF,A
197F 12198E     LCALL TRANS_DEL1
1982 04         INC A
1983 F599       MOV SBUF,A     ;LED3
1985 12198E     LCALL TRANS_DEL1
1988 02198A     LJMP OUT_TST
                NORMAL:   ::NORMAL DISPLAY
1988 C2A7       CLR P2.7    ;RESTORE ALARM & DISPLAYS
198D C2D3       CLR PSW.3   ;BANK0
198F C2D4       CLR PSW.4
1991 852899     MOV SBUF,CHG_VOL
1994 12198E     LCALL TRANS_DEL1
1997 759901     MOV SBUF,#01H
199A 12198E     LCALL TRANS_DEL1
199D 759902     MOV SBUF,#02H
19A0 12198E     LCALL TRANS_DEL1
19A3 8E99       MOV SBUF,RVENT
19A5 12198E     LCALL TRANS_DEL1
19A8 852399     MOV SBUF,LED1
19AB 12198E     LCALL TRANS_DEL1
19AE 852699     MOV SBUF,LED2
19B1 12198E     LCALL TRANS_DEL1
19B4 852599     MOV SBUF,LED3
19B7 12198E     LCALL TRANS_DEL1
                OUT_TST:
19BA 02133F     LJMP BLINK_BEEP
19BD 00         NOP

                TRANS_DEL1:   ;:DELAY 2.25MS,CC=80EH
19BE D51DFD     DJNZ DIVIDE1,TRANS_DEL1   ;COUNT 255
19C1 751DFF     MOV DIVIDE1,#0FFH   :RESET
19C4 D51EF7     DJNZ DIVIDE2,TRANS_DEL1   :COUNT 4
19C7 751E04     MOV DIVIDE2,#04H   :RESET
19CA 22         RET
19CB 00         NOP

                MAN_SW:   ::ON VALVES
19CC C0E0       PUSH ACC
19CE C0D0       PUSH PSW
19D0 C2A8       CLR EX0   :DISABLE INT
19D2 53A08F     ANL P2,#10001111B :ON VALVES
19D5 12156E     HOLDIT: LCALL SERVICE
19D8 30B2FA     JNB P3.2,HOLDIT
19DB D200       SETB WAIT
19DD D21E       SETB L16   :WAIT LED
19DF 852399     MOV SBUF,LED1
19E2 31BE       ACALL TRANS_DEL1
19E4 43A07C     ORL P2,#011100008   :OFF VALVES
19E7 752F00     MOV THREE_CYCLE,#00H
19EA D2A8       SETB EX0   :ENABLE INTO
19EC D0D0       POP PSW
19EE D0E0       POP ACC
```

```
19F0 32          RETI

                 ;%E
19F1             ENDS    :CODE SEGMENT
1000             END  BEGIN
```

| Symbol Name | Type | Value |
|---|---|---|
| ALARM | L | 112F |
| ALARM1 | L | 11AB |
| ALARM2 | L | 12A4 |
| ALM | B | 000C |
| ALM_TST | B | 000A |
| ANALOG | M | 0000 |
| AWP_AVG | D | 004C |
| AWP_LO | D | 0048 |
| AWP_MAX | D | 004D |
| BANK0 | U | 0000 |
| BCD_OUT0011 | L | 16DD |
| BCD_OUT0012 | L | 17E1 |
| BCD_OUT0013 | L | 192C |
| BEEP | B | 0014 |
| BEGIN | L | 1000 |
| BINARY_BCD | M | 0000 |
| BLINK_BEEP | L | 133F |
| BLINK_BEEP1 | L | 1623 |
| CALC_HUN0011 | L | 16B2 |
| CALC_HUN0012 | L | 1786 |
| CALC_HUN0013 | L | 1901 |
| CALC_ONE0011 | L | 16CC |
| CALC_ONE0012 | L | 17D0 |
| CALC_ONE0013 | L | 191B |
| CALC_PTP | L | 115D |
| CALC_TEMP | L | 16A3 |
| CALC_TEN0011 | L | 168F |
| CALC_TEN0012 | L | 17C3 |
| CALC_TEN0013 | L | 190E |
| CASE0 | L | 167D |
| CASE1 | L | 16F4 |
| CASE2 | L | 179F |
| CASE3 | L | 179D |
| CASE4 | L | 1726 |
| CASE5 | L | 17A1 |
| CASE6 | L | 186D |
| CASE61 | L | 165D |
| CASE7 | L | 1867 |
| CASE71 | L | 1661 |
| CASE8 | L | 1769 |
| CASE81 | L | 1765 |
| CASE9 | L | 1818 |
| CASE91 | L | 1665 |
| CASEA | L | 187A |
| CASEA1 | L | 1669 |
| CASEB | L | 186A |
| CASEB1 | L | 166D |
| CASEC | L | 1843 |
| CASEC1 | L | 1671 |
| CASED | L | 18D7 |
| CASED1 | L | 1675 |
| CASEE | L | 1955 |

```
CASEE1. . . . . L 1679
CASEF . . . . . L 165A
CHARGE. . . . . L 1184
CHGVOL_TBLE . . . . . L 178F
CHG_VOL . . . . . D 0028
CHK_AWP . . . . . L 1107
CHK_CHGTIM. . . . . L 11E6
CHK_DPTHRESH. . . . . L 1242
CHK_EOEXH . . . . . L 12A7
CHK_EOEXH1. . . . . L 123F
CHK_EXH . . . . . L 1100
CHK_HOLD. . . . . L 1429
CHK_LED21 . . . . . L 1358
CHK_LED22 . . . . . L 13A9
CHK_LED31 . . . . . L 1363
CHK_LED32 . . . . . L 1384
CHK_OFFTIM. . . . . L 15ED
CHK_PEAK. . . . . L 1150
CHK_SERPORT . . . . . L 1628
CHK_SIL . . . . . L 1134
CHK_VOL . . . . . L 11C6
CHK_VOL1. . . . . L 136E
CHK_VOL2. . . . . L 138F
CHK_WAIT. . . . . L 1308
CHK_WAIT1 . . . . . L 11E3
CLEAR . . . . . L 1407
CLK . . . . . B 0039
CLOG1 . . . . . B 0016
CLOG2 . . . . . B 0017
CLOG_HI . . . . . D 0046
CLOG_LO . . . . . D 0045
CONT. . . . . L 113C
CONT1 . . . . . L 1555
CONT2 . . . . . L 15CA
CONT3 . . . . . L 15D4
CONT4 . . . . . L 15DF
CONT5 . . . . . L 10D2
CONT6 . . . . . L 141A
DEC_HUN . . . . . D 0029
DEC_ONE . . . . . D 0028
DEC_TEN . . . . . D 002A
DEL1. . . . . B 0012
DELAY1. . . . . L 10C6
DELAY2. . . . . L 10C8
DELAY5. . . . . L 1122
DEL_4TENTHS . . . . . B 0013
DISPLAY_TEMP. . . . . L 16A7
DIV21 . . . . . B 0002
DIV22 . . . . . B 0008
DIV23 . . . . . B 0038
DIV24 . . . . . B 000E
DIVIDE1 . . . . . D 001D
DIVIDE2 . . . . . D 001E
END_DEL . . . . . L 10D0
EOT . . . . . L 10EB
```

```
EXH . . . . . B 0001
FIFO. . . . . M 0E84
FILTAWP_THRESH. . . . . I 0037
FILTDP_THRESH . . . . . I 008D
FIRST_SAMP. . . . . L 1220
FLO . . . . . L 15D8
FLOTIM. . . . . I 0008
FLOW. . . . . B 000F
FLO_ALM . . . . . L 160E
FLO_TH. . . . . I 002D
FLO_TST . . . . . L 1585
FLTFLO_AVG. . . . . D 0044
FLTFLO_LO . . . . . D 0040
FLTLD25 . . . . . D 0055
FLTLD50 . . . . . D 0056
FLTLD75 . . . . . D 0057
FLTLD_HUN . . . . . D 002C
FLTLD_ONE . . . . . D 002E
FLTLD_TEN . . . . . D 002D
FLT_LD. . . . . L 1225
GO_ON . . . . . L 1339
HEAT. . . . . B 003A
HEAT_CHK. . . . . L 1592
HIPRESS . . . . . L 11A3
HITEMP. . . . . L 157B
HI_TEMP . . . . . L 15AB
HOLD. . . . . B 0008
HOLDIT. . . . . L 19D5
HOLD_OFF. . . . . L 1708
HOLD_OUT. . . . . L 1716
INC3. . . . . L 1336
INIT1 . . . . . L 1032
INITIALIZE. . . . . L 1030
INSP. . . . . B 0015
JUMP_TBLE1. . . . . L 1639
L14 . . . . . B 001C
L15 . . . . . B 001D
L16 . . . . . B 001E
L17 . . . . . B 001F
L24 . . . . . B 0034
L25 . . . . . B 0035
L26 . . . . . B 0036
L27 . . . . . B 0037
L34 . . . . . B 002C
L35 . . . . . B 0020
LCD_VOL . . . . . L 1945
LED1. . . . . D 0023
LED2. . . . . D 0026
LED3. . . . . D 0025
MAIN_LOOP . . . . . L 10D8
MAN_SW. . . . . L 19CC
NEGFLO. . . . . L 154A
NEGSUM. . . . . D 0051
NEW_TEMP. . . . . L 1699
NEXT10001 . . . . . L 116D
```

| | | |
|---|---|---|
| NEXT10002 | L | 12C9 |
| NEXT10004 | L | 145D |
| NEXT10006 | L | 14A6 |
| NEXT10008 | L | 14EF |
| NEXT20001 | L | 1185 |
| NEXT20002 | L | 12E1 |
| NEXT20004 | L | 1474 |
| NEXT20006 | L | 148D |
| NEXT20008 | L | 1506 |
| NEXTSUB10011 | L | 1684 |
| NEXTSUB10012 | L | 1788 |
| NEXTSUB10013 | L | 1903 |
| NEXTSUB20011 | L | 16C1 |
| NEXTSUB20012 | L | 17C5 |
| NEXTSUB20013 | L | 1910 |
| NEXT_SAMP | L | 1524 |
| NOFLO | L | 15D0 |
| NOFLOTIM | I | 0032 |
| NOFLO_ALM | L | 15F7 |
| NOFLO_TH | I | 008C |
| NORMAL | L | 1988 |
| NOT_EQ | L | 1587 |
| OFFTIMER | D | 0015 |
| OFF_ALM | B | 000D |
| OFF_STATE | L | 1687 |
| ONTIMER | D | 0014 |
| OUT | L | 13EB |
| OUT1 | L | 133C |
| OUTPIP | L | 1952 |
| OUT_DISPLD | L | 1840 |
| OUT_TEMP | L | 1815 |
| OUT_TST | L | 198A |
| PATINSP_THRESH | I | 0008 |
| PAT_INSP | L | 12F6 |
| PEEP_AVG | D | 0063 |
| PEEP_LO | D | 0060 |
| PIP_AVG | D | 0058 |
| PIP_LO | D | 0058 |
| PIP_STORE | D | 004E |
| PIP_THRESH | I | 00E0 |
| POSSUM | D | 0050 |
| RAW_PRESS | R | 0003 |
| RCHG_TIM | R | 0001 |
| RDIV10 | R | 0002 |
| RDIV5 | R | 0003 |
| RESTORE_VOL1 | L | 17FF |
| RESTORE_VOL2 | L | 182E |
| RETURN | L | 1567 |
| RFLT_FLO | R | 0002 |
| RHOLD_TIM | R | 0007 |
| RNEB_FLO | R | 0004 |
| ROFF_TIM | R | 0005 |
| RON_TIM | R | 0004 |
| RSIL_TIM | R | 0006 |
| RTEMP | R | 0005 |

```
RUNNING_AVG . . . . . M 0000
RVENT . . . . . R 0006
RVENT_SIG . . . . . R 0001
SAMPLE. . . . . L 1435
SEE_LD. . . . . B 0011
SEE_PIP . . . . . B 0009
SEE_TEMP. . . . . B 0010
SEE_VENT. . . . . L 1743
SERVICE . . . . . L 156E
SET_CHGTIM. . . . . D 0019
SET_CLK . . . . . L 1413
SET_EXH . . . . . L 1117
SIL . . . . . B 0006
SPON_BR . . . . . B 0007
STRT_EXH. . . . . L 11AE
SW_OFF. . . . . L 15A1
TEMP. . . . . B 003B
TEMP_CONT . . . . . L 1573
TEMP_DEC. . . . . D 0012
TEMP_HI . . . . . I 00A0
TEMP_SET. . . . . D 0011
TEMP_STORE. . . . . D 001C
TEST0 . . . . . L 1298
TEST25. . . . . L 1286
TEST50. . . . . L 1274
TEST75. . . . . L 1258
THREE_CYCLE . . . . . D 002F
TIM_SAMP. . . . . L 13F3
TRANS_DEL . . . . . L 1625
TRANS_DEL1. . . . . L 198E
TST_BEEP1 . . . . . L 1394
TST_BEEP2 . . . . . L 13E6
TST_TEMP1 . . . . . L 1387
TST_TEMP2 . . . . . L 13D9
TST_VENTSEL1. . . . . L 1380
TST_VENTSEL2. . . . . L 13D1
TURN_OFF. . . . . L 139E
TURN_ON . . . . . L 134E
VENT_AVG. . . . . D 006C
VENT_HI . . . . . D 001B
VENT_LO . . . . . D 001A
VENT_LOW. . . . . D 0068
VENT_TBLE . . . . . L 175D
VEN_SEL . . . . . B 0004
VOL_CHG . . . . . B 0003
WAIT. . . . . B 0000

:%Z

00 Errors  (0000)
```

WHAT IS CLAIMED IS:

1. A nebulizer comprising:

a housing containing a reservoir for holding a liquid to be nebulized and an air space above the reservoir for holding aerosol;

means for generating said aerosol by nebulizing said liquid;

means for attaching said housing to a mechanical respirator having an inhalation phase, an exhalation phase, a gas flow passageway to a patient, and an external electrical signal source capable of generating a first electrical signal during said exhalation phase;

means responsive to said first electrical signal for introducing said aerosol into said gas flow passageway, such that said aerosol fills said gas flow passageway during a portion of said exhalation phase.

2. The nebulizer of Claim 1 further comprising means for monitoring the amount of said aerosol introduced into said gas flow passageway.

3. The nebulizer of Claim 1 wherein said mechanical respirator further being capable of generating a second electrical signal during said inhalation phase.

4. The nebulizer of Claim 3 wherein said aerosol generating means further comprising a plurality of nebulizer nozzles each having means for controlling the gas flow therethrough.

5. The nebulizer of Claim 4, wherein said introducing means further comprises:

a gas flow for directing compressed gas from a compressed gas source to each of said plurality of controlling means for said nebulizer nozzles; said gas flow means including means responsive to said first electrical signal for opening a conduit of said nebulizer nozzles and for closing the conduit to said nebulizer nozzles simultaneously or one at a time, in response to said second electrical signal.

6. The nebulizer of Claim 5 further comprising:

means responsive to said second electrical signal for generating a decreasing flow of gas; and

means for directing said decreasing flow of gas into said mechanical respirator.

7. A method of operating a nebulizer of the type having means for generating an aerosol and means for supplying said aerosol to a mechanical respirator having an inhalation phase, an exhalation phase and a gas passageway to a patient, and an external electrical signal source capable of generating a first electrical signal during said exhalation phase, method comprising:

generating said aerosol; and

introducing said aerosol into said gas passageway during a portion or all of the said exhalation phase.

8. The method of Claim 7 wherein said introducing step further comprising:

opening a valve, in response to said first signal, to introduce said aerosol from said nebulizer to said gas passageway.

9. The method of Claim 7 wherein said generating step further comprises:

entraining a liquid into a source of compressed gas to generate said aerosol, in response to said first signal and continuing until standardized volume of aerosol dose has been delivered.

10. The method of Claim 7 wherein said external electrical signal source is capable of generating a second electrical signal during said inhalation phase.

11. The method of Claim 10 further comprising:

ceasing the generation of said aerosol in response to said second electrical signal.

12. A nebulizer for use with a respirator means having an inhalation phase and an exhalation phase, a first tubing means connecting said respirator means with a patient wherein during said inhalation phase said respirator means is fluidically connected to said patient through said first tubing means for introducing breathing gas in said first tubing means into respiratory tract of the said patient, a second tubing means connecting said respirator means with said patient wherein during said exhalation phase said respirator means is fluidically connected to said patient through said second tubing means for receiving exhaled gas from said patient to said respirator means, said respirator means further

having means for generating a first electrical signal during said exhalation phase; said nebulizer comprising:

means for generating an aerosol;

aerosol connecting means for connecting said generating means to said first tubing means; and

means for introducing said aerosol into said first tubing means in response to and synchronized with said first electrical signal.

13. The nebulizer of Claim 12 further comprising:

housing means containing a reservoir for holding a liquid to be nebulized and an air space above the reservoir for holding said aerosol.

14. The nebulizer of Claim 13 wherein said aerosol connecting means connects said air space to said first tubing means.

15. The nebulizer of Claim 14 wherein said generating means comprising:

a plurality of nebulizing nozzles each having means for controlling the gas flow therethrough.

16. The nebulizer of Claim 15 wherein said respirator means for generating a second electrical signal during said inhalation phase.

17. The nebulizer of Claim 16 wherein said introducing means for all of said nebulizing nozzles, in response to said first electrical signal, de-

activates said controlling means, either simultaneously or one at a time.

18. The nebulizer of Claim 14 further comprising means for monitoring said aerosol introduced into said first tubing means.

19. The nebulizer of Claim 16 further comprising:

means for generating a decreasing flow of gas; and

means for directing said decreasing volume of gas into said second tubing means.

20. The nebulizer of Claim 12 wherein said means for generating said first electrical signal further comprises:

a filter pressure sensor for detecting the pressure differential in said second tubing means, and for generating a filter pressure signal in response thereto;

an airway pressure sensor for detecting the pressure in said first tubing means, and for generating an airway pressure signal in response thereto; and

means for receiving said filter pressure signal and said airway pressure signal and for generating said first electrical signal synchronized with the commencement of said exhalation phase.

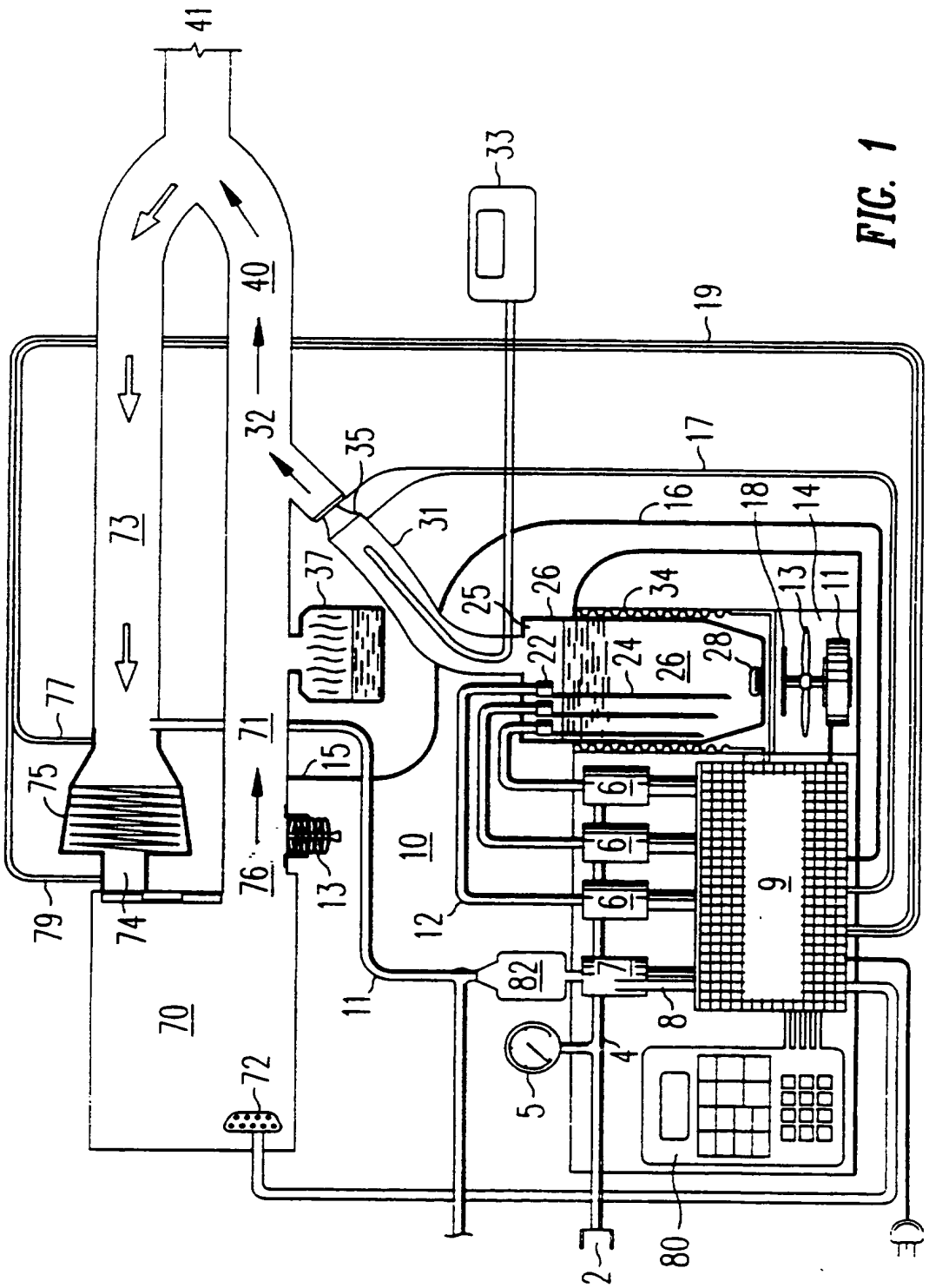



FIG. 1

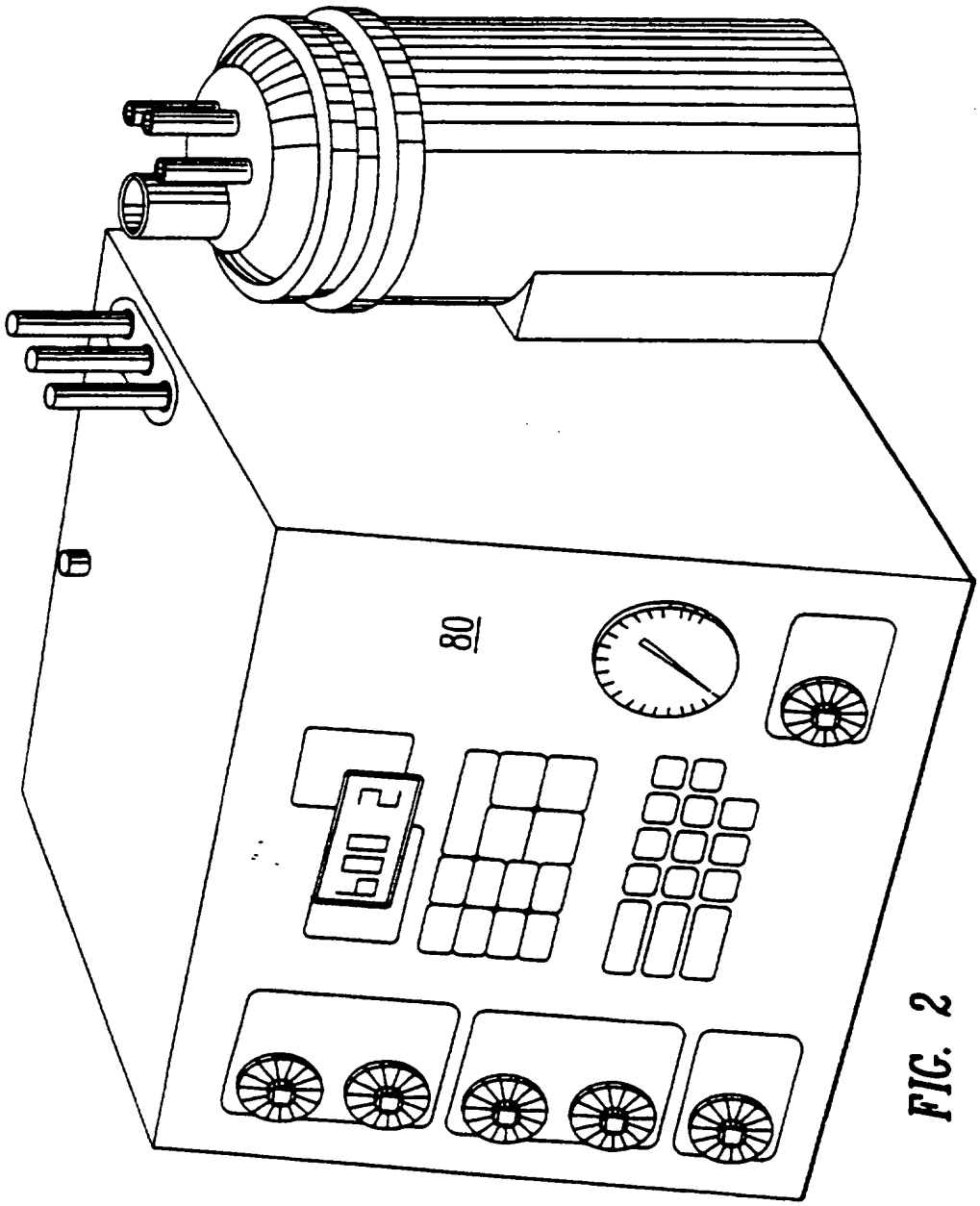



FIG. 2

# INTERNATIONAL SEARCH REPORT

International Application No. PCT/US92/00566

## I. CLASSIFICATION OF SUBJECT MATTER (if several classification symbols apply, indicate all) [6]

According to International Patent Classification (IPC) -- to both National Classification and IPC

I.P.C. (5): A61M 15/00, A61M 16/10, A62B 7/00, F16K 31/02
U.S. Cl. : 128/203.12, 204.21, 204.23, 204.26

## II. FIELDS SEARCHED

Minimum Documentation Searched [7]

| Classification System | Classification Symbols |
|---|---|
| U.S. | 128/200 14, 200.21; 203.12, 203.13, 203.14, 203.16, 203.17 203.26, 203.27, 204.17, 204.18, 204.21, 204.23, 204.26 |

Documentation Searched other than Minimum Documentation to the Extent that such Documents are included in the Fields Searched [8]

## III. DOCUMENTS CONSIDERED TO BE RELEVANT [9]

| Category * | Citation of Document, [11] with indication, where appropriate, of the relevant passages [12] | Relevant to Claim No. [13] |
|---|---|---|
| Y | US, A, 4,106,503 (ROSENTHAL et al) 15 AUGUST 1978 See entire document | 1-3,7-14,18-20 |
| Y | US, A, 4,832,014 (PERKINS) 23 MAY 1989 See entire document | 1-3,7-14,18-20 |
| Y | US, A, 4,197,843 (BIRD) 15 APRIL 1980 See entire document | 1-3,7-14,18-20 |
| Y | US, A, RESPIRATORY THERAPY EQUIPMENT (MCPHERSON) @1985, C.V. MOSBY CO., pp. 128-131, 158-163, 468-469, 476-479 & 442-443 | 1-3,7-14,18-20 |

* Special categories of cited documents: [10]

"A" document defining the general state of the art which is not considered to be of particular relevance

"E" earlier document but published on or after the international filing date

"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step

"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art.

"&" document member of the same patent family

## IV. CERTIFICATION

| Date of the Actual Completion of the International Search | Date of Mailing of this International Search Report |
|---|---|
| 27 APRIL 1992 | 27 MAY 1992 |
| **International Searching Authority** | **Signature of Authorized Officer** |
| ISA/US | KIMBERLY L. ASHER |

Form PCT/ISA/210 (second sheet) (Rev. 11-87)